University of Alberta

Simulation of Galaxy Formation

by

Robert John Thacker

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

Department of Physics

Edmonton, Alberta
Fall 1999

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisitions et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-46931-X

Canada

University of Alberta

Library Release Form

Name of Author: Robert John Thacker

Title of Thesis: Simulation of Galaxy Formation

Degree: Doctor of Philosophy

Year this Degree Granted: 1999

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

Robert J. Thacker

Robert John Thacker
Department of Physics
University of Alberta
Edmonton, AB
Canada, T6G 2J1

Date: 27th Aug. 1999

Wonder is the beginning of wisdom.
–Greek proverb.

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Simulation of Galaxy Formation** submitted by Robert John Thacker in partial fulfillment of the requirements for the degree of **Doctor of Philosophy.**

Dr D. Hube

Dr D. Page

Dr H. M. P. Couchman

Dr R. Sydora

Dr G. Swaters

Dr A. Evrard

Date: 27th Aug. 1999

Abstract

This thesis presents a detailed study of the simulation of galaxy formation in the cold dark matter (CDM) cosmology. Smoothed Particle Hydrodynamics (SPH) is used to follow the hydrodynamics within the simulations and an analysis of the performance of twelve different implementations of SPH is presented. Seven tests are used which are designed to isolate key hydrodynamic elements of cosmological simulations. It is shown that the artificial viscosity is the single most important factor in determining results. The way in which force symmetry is achieved in the equation of motion has a secondary effect. Most results favour a kernel symmetrization approach. A detailed description of a method for simulating the formation of individual galaxies is given. Gas regions which fall within temperature, density, self-gravity and convergent-flow criteria are assumed to form stars. A Lagrangian Schmidt Law is used to calculate the star formation rate. Feedback from supernovae is incorporated by returning thermal energy to the inter-stellar medium. Radiative losses are prevented from heated particles by adjusting the radiative cooling mechanism. The model is tested on isolated disc galaxies as well as galaxies formed from cosmological initial conditions. A discussion is presented on the implementation of 'HYDRA', a combined hydrodynamic and gravity N-body particle integrator, on shared-memory architectures with a symmetric multi-processor configuration (SMP). Parallelization is achieved using the directives in the OpenMP application program interface. The performance of the code is examined for up to 128 processors and excellent scaling is found, provided a large enough problem is studied. A high resolution simulation is presented which satisfies a number of criteria for accuracy defined in the SPH tests. Due to limitations of the parallel code it was not possible to integrate the simulation to the desired final epoch. At high resolution gas in-fall is seen to be highly unsmooth, and the gas appears to lose angular momentum more rapidly. Although higher resolution prevents the formation of very dense gas cores it does not entirely prevent the condensation of cold gas. Prospects for the future of the simulation field, and the CDM model of structure formation are given.

I dedicate this thesis to Linda Campbell, without whose love, support and understanding, things would have been so much harder.

Acknowledgements

Cosmology once lent itself to those who wished to study in solitude. This is no longer the case. Looking back over the five years of work that consitute this PhD there have been a number of influences that have enriched, altered and directed my path.

I must admit a great debt of gratitude to Dr Peter Coles. Although Peter has not contributed directly to this thesis it was his enthusiasm and inspiration that planted the seeds of my own interest. Ultimately I have him to thank for giving me the guts to give up a well paying job and dive into the swirling waters of cosmology!

Professors Don Page, Bruce Campbell and John Beamish are owed thanks. Bruce for giving a kick up the backside when I needed it and Don and John for supporting me through some difficult times (both financial and academic). Without Bruce's constant nudging I doubt I would have had the guts to leave Alberta and visit Western. Don has had the unpleasant(?) task of trying to supervise an absentee graduate student. Lynn Chandler has also run around for me on a number of occasions, coping with the U of A administrative work that I should have been doing, but couldn't. Thanks are also extended to the staff and faculty at Western for accomodating me during my two year visit. I also thank the staff and grad students of the University of Waterloo Biology department for allowing me to use their computers during final preparation of the manuscript.

Perhaps my largest token of thanks goes to my co-supervisor Professor Hugh Couchman. Hugh must take all the credit for helping me to realise a lot of ideas, and for being prepared to take yet another grad student under his wing. I have benefitted in no small part from him presenting himself as someone to bounce ideas off, and his never ending knack for redirecting and offering alternatives when things seemed to come to a dead end, has saved me countless hours of frustration.

I must also thank Dr Frazer Pearce, Dr Peter Thomas, Dr Eric Tittley and Todd Fuller for many discussions and numerous emails that have contributed to my understanding. Thanks to Prof. Matthias Steinmetz for providing his PPM results for the 'Evrard' collapse.

Jimmy Scott of SGI Canada deserves a big thank you. Jimmy helped secure time on Origin 2000 systems at the SGI Eagan Supercomputing Center. Without this more than generous grant of run time many of the results presented in the later chapters of this thesis would not have been possible. Also Dr. Stuart Rankin, system adminstrator for the UK-CCC 'COSMOS' supercomputer, provided invaluable help.

I thank the Canadian Commonwealth Scholarship Plan for providing support over the first four years and the University of Alberta for granting me a Dissertation Fellowship.

Last but by no means least, I thank Linda Campbell. Aside from her unending supply of love and affection, Linda also drafted a number of the figures in this thesis. I also must acknowledge that she made me realise just how lucky I am to able to do what I love doing, and helped me to rediscover the beauty in it when things became tedious.

Contents

1 Introduction 1
- 1.1 Cosmology . . . 1
- 1.2 Galaxies, COBE and Cold Dark Matter . . . 2
- 1.3 Useful mathematical results from cosmology . . . 3
  - 1.3.1 Jeans' Instability Theory: Static Background Analysis . . . 3
  - 1.3.2 Jeans' Instability Theory: Expanding Background Analysis . . . 5
  - 1.3.3 Zel'dovich Approximation . . . 7
  - 1.3.4 Relativistic Collapse Model . . . 8
  - 1.3.5 Statistics: Correlation Functions, Autocovariance and the Power Spectrum . 9
- 1.4 Galaxy formation via analytic and semi-analytic methods . . . 12
  - 1.4.1 An analytic model of galaxy formation . . . 12
  - 1.4.2 Semi-analytic models of galaxy populations . . . 17
- 1.5 The utility of simulation . . . 18
- 1.6 Particle simulation methods . . . 19
  - 1.6.1 Particle-Particle method . . . 19
  - 1.6.2 Particle-Mesh method . . . 19
  - 1.6.3 Particle-Particle, Particle-Mesh method . . . 20
  - 1.6.4 Barnes-Hut 'Tree' method . . . 20
  - 1.6.5 Smoothed Particle Hydrodynamics . . . 21
- 1.7 Progress to date of the simulation field and comparison to observation . . . 22
  - 1.7.1 Results: Problems inherent in the standard CDM picture . . . 22
  - 1.7.2 Results: Problems inherent in the numerics . . . 22
  - 1.7.3 Successes & Failures . . . 23
  - 1.7.4 A list of outstanding issues in galaxy formation . . . 25
- 1.8 Layout of this thesis . . . 26

2 A detailed examination of Smoothed Particle Hydrodynamics 27
- 2.1 Introduction . . . 27
- 2.2 Implementations of SPH . . . 29
  - 2.2.1 Features common to all implementations . . . 29
  - 2.2.2 An improved first-order smoothing-length update algorithm . . . 30
  - 2.2.3 Equations of motion . . . 33
  - 2.2.4 Internal energy equation . . . 34
  - 2.2.5 Artificial viscosity algorithms . . . 35
- 2.3 Summary of implementations . . . 36
- 2.4 Shock tube: Summary . . . 37
- 2.5 Drag Test: Summary . . . 37
- 2.6 Cooling test: Summary . . . 37
- 2.7 Spherical collapse . . . 38
  - 2.7.1 Units and initial conditions . . . 38
  - 2.7.2 System evolution . . . 39
  - 2.7.3 Results from the 485 particle collapse . . . 39
  - 2.7.4 Effect of numerical resolution . . . 39
  - 2.7.5 Summary . . . 42
- 2.8 Rotating cloud collapse . . . 42
  - 2.8.1 Initial conditions . . . 42
  - 2.8.2 Non-implementation-specific results . . . 45
  - 2.8.3 Implementation-specific results . . . 47
  - 2.8.4 Summary . . . 49
- 2.9 Angular momentum transport . . . 49
  - 2.9.1 General evolution properties . . . 50

2.9.2 Implementation-specific results . . . 50
2.9.3 Summary . . . 51
2.10 Cosmological simulation . . . 51
2.10.1 Initial Conditions . . . 51
2.10.2 Extraction of glob properties . . . 52
2.10.3 Results of cosmological test . . . 53
2.10.4 Summary . . . 55
2.11 Conclusions . . . 55

3 Incorporating star formation and feedback into simulations of galaxy formation 59
3.1 Introduction . . . 59
3.2 Numerical Method . . . 61
3.3 Star Formation Prescription . . . 61
3.3.1 Implementation details . . . 61
3.3.2 Returning feedback energy . . . 62
3.3.3 Comparison of methods . . . 64
3.3.4 Parameter space of the algorithm . . . 66
3.4 Application to isolated 'realistic' models . . . 68
3.4.1 Milky Way prototype . . . 68
3.4.2 Dwarf prototype . . . 74
3.5 Cosmological simulation . . . 76
3.5.1 Initial conditions . . . 76
3.5.2 Simulation Parameters . . . 78
3.5.3 System evolution without feedback . . . 79
3.5.4 Caveats . . . 80
3.5.5 Effect of feedback on SFR and morphology . . . 80
3.5.6 Halo profiles . . . 81
3.5.7 Rotation curves and angular momentum . . . 83
3.5.8 Auxiliary simulations . . . 89
3.6 Summary and Discussion . . . 91

4 Parallel computation in simulation of galaxy formation 94
4.1 Introduction . . . 94
4.2 CPU architectures and parallel programming paradigms . . . 96
4.2.1 Reduced Instruction Set Computer processors . . . 96
4.2.2 Architecture of the SGI Origin 2000 supercomputer . . . 96
4.3 Detailed breakdown and optimization of the serial algorithm . . . 97
4.3.1 Review of the serial algorithm . . . 97
4.3.2 Changes to artificial viscosity, equations of motion and energy . . . 103
4.3.3 Revised time-step criteria . . . 103
4.3.4 New smoothing length update algorithm . . . 104
4.3.5 Test-bed simulation . . . 104
4.3.6 Optimizing the memory access of the serial code . . . 105
4.4 Parallelization of HYDRA on shared memory multiprocessors . . . 107
4.4.1 Load balancing options provided by the OpenMP standard . . . 108
4.4.2 Data Geometry . . . 110
4.4.3 Particle velocity and position updates (updaterv) . . . 110
4.4.4 Periodic mass assignment (mesh) . . . 110
4.4.5 Fast Fourier transform (four3m) . . . 114
4.4.6 Periodic Fourier convolution (cnvl) . . . 114
4.4.7 Force interpolation (mesh) . . . 114
4.4.8 Pair-wise forces (shgravsph) . . . 115
4.4.9 Refinement placing (refine) . . . 116
4.4.10 Distribution of refinements (force) . . . 116
4.4.11 Isolated mass assignment (imesh) . . . 116
4.4.12 Isolated Fourier convolution (icnvlt) . . . 116
4.4.13 Isolated force interpolation (imesh) . . . 117
4.5 Performance of the parallel code . . . 117
4.5.1 Simulations with homogeneous particle distributions . . . 117
4.5.2 Simulations with inhomogeneous particle distributions . . . 121
4.6 Summary and discussion . . . 123

5 Galaxy formation in hierarchical clustering **128**
5.1 Introduction . . . 128
5.2 Initial conditions . . . 129
5.2.1 Low resolution dark matter simulation . . . 129
5.2.2 Selection of resimulation halo . . . 129
5.3 Simulation algorithm . . . 133
5.3.1 Parallelization of the star formation algorithm . . . 133
5.3.2 Refinement placing . . . 134
5.3.3 Inefficiencies in the algorithm . . . 134
5.4 Results . . . 134
5.4.1 Morphology and the effect of feedback . . . 136
5.4.2 Angular momenta of the dwarfs and main halo . . . 136
5.4.3 Halo and glob mass multiplicity functions . . . 144
5.4.4 Star formation rate . . . 144
5.4.5 Halo density profile . . . 148
5.4.6 Overmerging . . . 149
5.4.7 Halo temperature profile . . . 149
5.5 Conclusion . . . 152

6 Conclusion **155**
6.1 Effect of different implementations of SPH . . . 155
6.2 Modelling star formation and feedback . . . 155
6.3 Parallel methods . . . 156
6.4 High resolution studies . . . 157
6.5 Future work . . . 158
6.5.1 Achieving a better understanding of numerical resolution . . . 158
6.5.2 Improvements to the simulation algorithm . . . 159
6.5.3 Incorporating more realistic physics . . . 159
6.5.4 Prospects for galaxy formation in the CDM model of structure formation . . . 159
6.5.5 Prospects for the simulation field . . . 160

A A note regarding the artificial viscosity and energy equations in comoving coordinates **161**

List of Figures

1.1 Comparison of the density contrast in the full non-linear solution compared to that predicted by linear perturbation theory. . . . 10
1.2 Plot of the cooling function $\Lambda(T)$, normalized so that $\Lambda(T)n^2$ is the power output per unit volume. . . . 13
1.3 Plot of the parameter space of collapsing perturbations, including the effect of gas cooling. . . . 15

2.1 Improvement in the neighbour counts as each component of the new algorithm is added. 31
2.2 Weighting function compared to simple weighted averages. . . . 33
2.3 Convergence of energy values with particle number for the Evrard collapse test using version 12. . . . 38
2.4 Radial profiles for the 30976 and 485 particle Evrard collapses at t=0.80. . . . 40
2.5 Thermal energy plot for the 485 particle evrard collapse, comparing versions 1, 3, 7 and 12. . . . 42
2.6 N=485 Evrard collapse energy difference profiles. Plots display the difference (in normalised units) between version 12 and the version under consideration. . . . 43
2.7 N=4776 Evrard collapse energy difference profiles. Plots display the difference (in normalised units) between version 12 and the version under consideration. . . . 44
2.8 Thermal energy plot (in normalised units) for the 4776 particle Evrard collapse, comparing versions 1, 3, 7 and 12. . . . 45
2.9 Evolution of gas and dark matter in the 2×1736 particle collapse for version 10. . . . 46
2.10 Comparison of morphology for the $N = 1736$ run under varying h_{min}, at $t = 128$. . . . 47
2.11 Comparison of gas morphology for 2×1736 particle collapse for different SPH implementations. . . . 48
2.12 Rotation curves from the rotating cloud collapse for four different implementations. . . 50
2.13 Ratio of thermal energy to initial mechanical energy versus time for 5 different implementations. . . . 51
2.14 Fraction of gas mass above 30,000 K versus time in the rotating cloud collapse. . . . 52
2.15 Comparison of gas halo size in the rotating cloud collapse at $t = 128$ for versions 1 and 12. . . . 53
2.16 Time evolution of the half-mass radius, half-AM radius and the ratio of the two in the disk stability test. . . . 54
2.17 The mass multiplicity function in the cosmological simulations for different implementations. . . . 56
2.18 Object-by-object comparison of masses from different implementations in the cosmological simulation. . . . 57

3.1 Evolution of the cooling density and the SPH density following a single feedback event in the ESna scheme. . . . 64
3.2 Comparison of different feedback schemes using radial temperature profiles. . . . 65
3.3 Effect of feedback scheme on the time-step selection in the simulation. . . . 66
3.4 Dependence of the SFR on the c^* parameter in a model with no feedback. . . . 68
3.5 Dependence of the SFR on the $t_{1/2}$ and e^* parameters (for the ESa algorithm). . . . 70
3.6 Morphology of Milky Way simulations at t=323 Myr, comparing the effect of different feedback implementations. . . . 71
3.7 SFRs for the Milky Way prototype (time averaged over 160 time-steps to show trend). 72
3.8 Rotation curves and radial velocity dispersions for the Milky Way prototype at t=323 Myr, for the NF, SPa, ESa and TS runs. . . . 73
3.9 z-coordinate of gas particles in the NGC 6503 simulation sorted into bins of width 0.150 kpc ($h_{min}/2$). . . . 75

3.10 Comparison of rotation curves and radial velocity dispersions for the NGC 6503 dwarf simulation at t=580 Myr. 76
3.11 SFRs for the dwarf prototype, time averaged to show trend. 77
3.12 Layering of initial conditions of intial conditions used in the cosmological simulations. 78
3.13 Integrated SFRs for the entire high resolution region for the cosmological simulations. 82
3.14 Radial temperature profile for the NF,SPa and ESa runs. 83
3.15 Density profile for the NF run in the cosmological simulation. 84
3.16 Rotation curves for the NSF, TSa, ESa and SPa runs in the disc formed in the cosmological simulation. 85
3.17 Radial velocity dispersions in the gas disc formed in the cosmological simulation. . . 86
3.18 Specific angular momenta versus mass for different components of the system for a number of different feedback algorithms. 87
3.19 Plot of specific angular momentum versus radius, showing the raw particle data versus $|L_z|$ calculated from the rotation curve at $z = 1.09$ 88
3.20 SFRs for the auxiliary cosmological simulations. 90

4.1 Overlay of the chaining mesh on top of the potential mesh to show spacing and the search radius of the short range force. 98
4.2 Call tree of the serial HYDRA algorithm. 100
4.3 Comparison of the 'Santa Barbara' cluster simulation radial profiles produced by the updated SPH implementation compared to the old version. 106
4.4 Execution time for the Santa Barbara cluster simulation for codes using different list structures. 108
4.5 Call tree of the OpenMP based parallel HYDRA algorithm. 109
4.6 Cell grouping and sorting in the 1×1 load balancing scheme. 111
4.7 Cell grouping and sorting in the 2×2 configuration scheme. 112
4.8 Example of a search through the concurrency matrix to determine which loads can be performed concurrently. 114
4.9 Maximum parallel scaling possible for the parallel mass assignment algorithm which uses slabs and ghost cells. 115
4.10 Relative scaling of the test simulations. The cluster simulation is replicated either 8 or 64 times to form the larger simulations. 119
4.11 Execution time in seconds for the most computationally expensive subroutines versus the number of PEs for the unclustered 128^3 simulation. 120
4.12 Scaling of the top level grid, large level 1 refinements and refinement farm at the end of the 128^3 simulation. 121
4.13 Parallel speed-up for the top level **shgravsph** routine. 123
4.14 Parallel speed-up for the top level **mesh** routine. 124
4.15 Parallel speed-up for the top level **list** routine. 125
4.16 Parallel speed-up for the top level **reorder** routine. 126
4.17 Relative cost of the top level grid, large level 1 refinements and refinement farm compared for different size simulations and different numbers of PEs. 127

5.1 Power spectra for the glass noise, Poisson noise and the applied perturbation spectrum. 130
5.2 4 panel zoom in on the resimulation halo in the 100^3 simulation to show environment. 132
5.3 Placement of the adaptive refinements within the entire simulation box at $z = 2.2$. 133
5.4 4-panel plot showing the evolution of the comoving density from $z = 10$ to the final epoch $z = 2.16$. 137
5.5 4-panel plot showing the evolution of the physical temperature from $z = 10$ to the final epoch $z = 2.16$. 138
5.6 Projections of the gas temperature and density at $z = 2.16$. 139
5.7 4-panel zoom in on the gas in the high resolution region to show environment. 140
5.8 Projections of the gas and star particles in the two largest dwarf systems at $z = 2.2$. 141
5.9 Three projections of a gaseous tidal tail formed by the interaction of two dwarf galaxies. 142
5.10 Plot of the z-component of the angular momentum of the dwarf systems, at $z = 2.2$, compared to the $L_z = Rv_c$ prediction. 143
5.11 Plot of the z-component of the angular momentum of the main halo at $z = 2.16$ compared to the $L_z = Rv_c$ prediction. 144
5.12 Specific angular momenta for the gas cores, hot halo, dark matter halo and star particles, within r_{200} of the main halo. 145
5.13 Groups found using the 'friends-of-friends' group identification algorithm at $z = 10, 5, 2.16$. 146
5.14 The mass multiplicity function for halos and globs in the simulation, corresponding to the groups shown in Figure 5.13. 147

5.15 SFR of the high resolution run integrated over the entire gas sector of the simulation. 148
5.16 Density profiles for the dark matter and gas in the main halo. The profiles are constructed using 208 particle Lagrangian bins. 150
5.17 Projection of the dark matter, gas and stellar distributions in the main halo at $z = 2.16$. 151
5.18 Distribution of the gas within the temperature-density plane at $z = 2.16$. 152
5.19 Temperature profile for the main halo at $z = 2.16$, including $\pm 0.5\sigma$ errors. 153

List of Tables

1.1 Names of common power spectrum indices. 11
1.2 Cooling times and collapse times for primordial clouds characterized by redshift, overdensity, $\Omega_0 h^2$ and baryon fraction. 16

2.1 Summary of the different implementations of SPH considered in the investigation. . 37
2.2 Results of Evrard collapse test. 41
2.3 Results of rotating cloud collapse test. 49
2.4 Properties of the cosmological test runs by version. 55
2.5 Qualitative summary of the strengths and weaknesses of each implementation as derived from the test results. 58

3.1 Summary of star formation parameter space simulations. 67
3.2 Summary of the main simulations using realistic models. 69
3.3 Summary of the properties of cosmological simulations at z=1.09. 80

4.1 Overall parallel scaling for simulations with comparatively homogeneous particle distributions. 118
4.2 Overall parallel scaling for simulations with inhomogeneous particle distributions. . . 122

5.1 Summary of simulation parameters 131

Chapter 1

Introduction

"To look backward for a while is to refresh the eye, to restore it, and to render it the more fit for its prime function of looking forward."

–Margaret Fairless Barber

1.1 Cosmology

Cosmology is one of the most audacious of the physical sciences. By utilizing data recorded on our planet, or at least local to it, cosmologists attempt to reconstruct the structure and past history of the Cosmos. Such an endeavour requires highly predictive theories since without them cosmology would be a morass of opinions and beliefs. Fortunately theoretical physics, backed by confirmation from experimental results, provides the necessary theories. However, whether experimental laws derived in the modern Universe should apply earlier in its lifetime, is not predicted by Science. Consequently Cosmology adopts what Wheeler calls a "Strong Bargaining Position", in that it is believed the theories derived for our current Universe must apply at all times in its evolution. Common sense indicates that without adopting this position scientific investigation would be almost impossible.

Although the subject of debate, modern cosmology is largely believed to have begun with Hubble's (1929) discovery of a linear relationship between redshift and distance. This discovery was motivated in part by de Sitter's (1917) note that redshift would be apparent in his *stationary* solutions. Hubble's discovery, in concert with his earlier work on the cosmological distance scale, rapidly changed the observational view of the Universe. The weight of these new discoveries, in conjunction with the growing body of theoretical knowledge, set a revolution in motion. The verification by Hubble of a general expansion law largely put to an end the debate of whether the Universe was static or not. Consequently Einstein's (1917) world model, which had been shown to be unstable by Eddington (1924), could be discarded and the expanding solutions of Friedmann (1917) and Lemaitre (1927) became more accepted. Note that Hubble's contribution was not limited to just the redshift-distance relation and the distance scale, he also conducted the first analysis of detailed galaxy surveys to show the number count as a function of magnitude relationship. The results from this analysis, namely that there appears to be no edge to the galaxy distribution and the counts obey the result for a homogeneous Universe, were the first to support Einstein's postulate of large-scale homogeneity. He also invented the Hubble (1926) Sequence for classifying galaxies which remains in use.

As we approach the beginning of the third millennium, cosmology may well be facing a rebirth. Until recently the field was largely starved for data and theories became ever more elaborate in the absence of data to constrain them. The first step towards rectifying this situation can be viewed as the Cosmic Microwave Background Explorer (COBE, Smoot *et al.*, 1992) experiment. Results from the COBE satellite, detailing fluctuations in the relic cosmic microwave background (CMBR), firmly pin down the large-scale structure in the Universe. In concert with a number of redshift surveys a broad picture of the large-scale structure of the Universe has begun to develop albeit missing data in the band between the largest redshift surveys and the scales probed by COBE. The new generation of galaxy surveys (2 Degree Field (2DF), Deep Extra-galatic Evolution Probe (DEEP) and the Sloan Digital Sky Survey) will map our local Universe with exquisite precision and also provide the first statistically complete sample of galaxies from an epoch when the Universe was 30% of its current age. At the same time as these surveys are being conducted, the Microwave Anisotropy Probe (MAP) and Planck Surveyor Satellite will measure the fluctuations in the CMBR to far higher precision

than in the COBE experiment. In concert with results from the Supernovae Cosmology project the global geometry of the Cosmos, and the division of the matter content within it, should be known within ten years. This second revolution is perhaps the reverse of the first as Cosmology will evolve from a theory driven subject back to a data driven one. The truly vast amounts of data resulting from the new experiments will tax theorists greatly.

1.2 Galaxies, COBE and Cold Dark Matter

Galaxies are uniquely interesting objects in cosmology since they are the milestones by which we can measure the Universe. They hold the position of being the smallest—and yet most complex—objects of interest to Cosmology. As a consequence of their size, physical phenomena other than gravity, such as gas dynamics, contribute to their formation and evolution. Thus the study of galaxy formation does not share the 'simplicity', and perhaps 'elegance', of related fields like large scale clustering. Given these considerations, the importance of understanding galaxy formation has perhaps rendered it the 'Holy Grail' of modern cosmology.

In the wider context of the "Standard Model" of Cosmology (Weinberg, 1972) the initial conditions for structure formation are widely believed to be a set of adiabatic (curvature) perturbations. Such fluctuations in the local energy density (albeit measured in particle number) are the same for all species. The fluctuations, which have been measured by the COBE satellite at an approximate redshift[1] of $z = 1000$, are a relic of some earlier dynamical phase (typically viewed to be the 'Inflationary Epoch', see Kolb and Turner, 1990). Relative to the background energy density the sizes of these fluctuations are miniscule. Measured in terms of the departure from anisotropy, the deviation is 1 part in 10^5, after our motion relative to the last scattering surface is subtracted. That such small perturbations should grow into the structure seen today is evidence of both the age of the Universe, and the long range nature of the gravitational force.

Our Universe cannot consist solely of baryonic matter if the adiabatic perturbation model is correct. The primary reason for this is that the pressure support of gas would prevent the formation of small scale structures, *i.e.* galaxies, before galaxy cluster size objects form. This is entirely contrary to observation. The assumption of a critical-density Universe continues to be more of a theoretical imperative than an observational certainty. Consequently, the failure of the purely baryonic adiabatic model spawned 'dark matter' models where the effect of the baryons is reduced. These models are differentiated into 'hot' models, where the 'dark' particles are relativistic at the matter-radiation decoupling epoch, and 'cold' models, where they are not. The cold dark matter (CDM, Blumenthal *et al.*, 1984) model comprising a critical-density Universe with a baryon fraction of 5-10% has become the 'straw man' of cosmology. It has many desirable features, but fails to adequately fit a number of observations (Peebles, 1993) when constrained by the CMBR fluctuations. The adiabatic hot dark matter model (HDM, Bond and Szalay, 1983) is largely discredited since it does not provide a mechanism to form galaxies (the first structures to form are of a size comparable to galaxy clusters). Because the CDM model offers a number of attractive features close to those observed today (*i.e.* formation of structure in a hierarchical 'bottom up' fashion), it continues to be a popular theory. Consequently, this thesis adopts solely the CDM model of structure formation as its starting point.

The main competitor to the CDM model of galaxy formation is the 'topological defect' picture. In this paradigm, spatial defects occur as a primordial field loses spatial symmetry and different sectors of the Universe acquire different values for the field. This process is akin to phase changes where, in passing from a liquid to a solid phase, a media acquires a preferred direction prescribed by the solid lattice. Defects then occur at the boundary of regions with different lattice alignments. The topology of the defects may be 0,1,2, or 3-dimensional (known as monopoles, strings, walls and textures, respectively). Defects continue to be a possible model for structure formation albeit in combination with other structure formation scenarios (*i.e.* strings plus hot dark matter, Abel *et al.*, 1998). Recent theoretical work suggests a large amount of energy from string annihilation can be deposited in particles which should be detectable as cosmic rays (while earlier work suggested this energy should be radiated as gravitational waves). If this picture is correct then measurements of the cosmic ray flux should be able to place a bound on defect models (Hindmarsh, 1998). It is hoped that the next generation of CMBR experiments, measuring the sub-degree scales, will be able to differentiate between this model and the adiabatic perturbation scenario (Turok, 1996).

[1] Where z is defined in terms of the scale factor, $a(t)$, such that $z = a(t_0)/a(t_e) - 1$ where t_0 is the current epoch and t_e is the emission epoch.

1.3 Useful mathematical results from cosmology

The Friedmann-Robertson-Walker line element for a homogeneous, isotropic and expanding cosmology is given by

$$ds^2 = c^2dt^2 - R^2(t)[\frac{dr^2}{1-kr^2} + r^2(d\theta^2 + \sin^2\theta d\phi^2)], \tag{1.1}$$

where $R(t)$ is the scale factor, determining the expansion of the 3-d world sheet, and $k \in \{-1, 0, 1\}$ determines whether the topology is open or closed.

The energy momentum tensor for a perfect fluid is given by

$$T_{\mu\nu} = (\rho + P)u_\mu u_\nu - pg_{\mu\nu}. \tag{1.2}$$

Substitution of this into the Einstein field equation

$$G_{\mu\nu} = 8\pi G T_{\mu\nu} + \Lambda g_{\mu\nu} \tag{1.3}$$

and use of the FRW line element leads to, after much manipulation, the Friedman equations for the scale factor $R(t)$,

$$H^2 \equiv \left(\frac{\dot{R}}{R}\right)^2 = \frac{8\pi G}{3}\rho - \frac{k}{R^2} + \frac{\Lambda}{3},$$

$$\frac{\ddot{R}}{R} = -\frac{4\pi G}{3}(\rho + 3P) + \frac{\Lambda}{3}. \tag{1.4}$$

If the effect of pressure is ignored and the cosmological constant is neglected, then the above equations correspond exactly to a Newtonian model for expanding 'dust'. This follows since the second Friedman equation for the 'acceleration' of the scale factor may be recast as

$$m\ddot{r} = -\frac{4\pi G\rho r m}{3} \tag{1.5}$$

where $r = R(t)s$ and s is a constant comoving coordinate. This equation trivially describes the motion of a fluid element of mass m under the influence of a sphere of radius r and density ρ.

Since $H(t) = \dot{R}/R$, the first Friedmann equation can be rearranged to give

$$E = T + V = \frac{1}{2}m\dot{r}^2 - \frac{4\pi G\rho}{3}mr^2, \tag{1.6}$$

where $E = -mks^2/2$, which is interpreted as the total energy of the fluid element. The potential energy is simply that of a fluid element with velocity $\dot{r}$ on the surface of a sphere of radius r and density ρ. The value of E then determines whether the radial motion continues *ad infinitum* or reaches a maximal expansion point and begins contracting.

The correspondence presented simplifies the analysis since it is unnecessary to integrate fully relativistic dust models. However, even without this exact correspondence, the weak field and slow motion analysis, valid for systems where $v \ll c$ and gravitational fields are weak, would still ensure that the Newtonian model can be used to a good approximation locally.

1.3.1 Jeans' Instability Theory: Static Background Analysis

The foundation of structure formation, at least on galaxy size scales, is the Jeans' Instability (Jeans, 1928). The theory describes the evolution of a collisional fluid under its self-gravity. To emphasize the physics of the model, the non-expanding version of the theory is presented first, and then generalized to to an expanding background.

The evolution of the fluid is governed by four equations (Lagrangian and Eulerian representations of the convservation type equations are given),

1. the continuity equation,

$$\frac{d\rho}{dt} + \rho\nabla.\mathbf{v} = 0, \quad \frac{\partial\rho}{\partial t} + \nabla.(\rho\mathbf{v}) = 0 \tag{1.7}$$

2. the Euler equation,

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho}\nabla P - \nabla\phi, \quad \frac{\partial\mathbf{v}}{\partial t} + (\mathbf{v}.\nabla)\mathbf{v} + \frac{1}{\rho}\nabla P + \nabla\phi = 0 \tag{1.8}$$

3. the Poisson equation,

$$\nabla^2\phi = 4\pi G\rho \tag{1.9}$$

4. the entropy conservation equation,

$$\frac{ds}{dt} = 0, \quad \frac{\partial s}{\partial t} + \mathbf{v}.\nabla s = 0. \tag{1.10}$$

where the conservation of entropy is a result of ignoring dissipation, viscosity and thermal conductivity (*i.e.* an ideal fluid). The dynamical system is closed by the equation of state $P = P(\rho, s)$. Alternatively, the entropy equation can be substituted with the conservation of energy equation,

$$\frac{du}{dt} = -\frac{P}{\rho}\nabla.\mathbf{v}, \quad \frac{\partial u}{\partial t} + (\mathbf{v}.\nabla)u + \frac{P}{\rho}\nabla.\mathbf{v} = 0, \tag{1.11}$$

and the equation of state is then $P = P(\rho, u)$.

The (Eulerian) perturbative analysis proceeds by considering small changes about a static solution. Note that placing $\nabla\phi = 0$ into the Poisson equation results in a density $\rho = 0$, *i.e.* a genuine static solution only exists for the vacuum. Nonetheless this point is overlooked in the static analysis, which has become known as the Jeans' Swindle.

Given the existence of a static solution, the following prescription is used to perturb about it,

$$\rho = \rho_0 + \delta\rho, \tag{1.12}$$

$$\mathbf{v} = \mathbf{0} + \delta\mathbf{v}, \tag{1.13}$$

$$P = P_0 + \delta P, \tag{1.14}$$

$$s = s_0 + \delta s, \tag{1.15}$$

$$\phi = \phi_0 + \delta\phi. \tag{1.16}$$

Assuming an equation of state for the fluid, $P = P(\rho, s)$, small variations in the pressure may be written,

$$\delta P = \left(\frac{\partial P}{\partial \rho}\right)_s \delta s + \left(\frac{\partial P}{\partial s}\right)_\rho \delta\rho = v_s^2\delta s + \left(\frac{\partial P}{\partial s}\right)_\rho \delta\rho, \tag{1.17}$$

where v_s^2 is the adiabatic sound speed of the fluid. This relation allows substitution for $\delta\rho$ in any equation. Upon substituting into equations (1-4) the resulting equations, describing the perturbation evolution, admit plane wave solutions,

$$\delta\begin{pmatrix}\rho\\ \mathbf{v}\\ \phi\\ s\end{pmatrix} = \begin{pmatrix}D\\ \mathbf{V}\\ \Phi\\ \Sigma\end{pmatrix} e^{i\omega t}e^{i\mathbf{k}.\mathbf{r}}, \tag{1.18}$$

provided that

$$\frac{\omega D}{\rho_0} + \mathbf{k}.\mathbf{V} = 0, \tag{1.19}$$

$$\omega\mathbf{V} + \frac{v_s^2 D}{\rho_0}\mathbf{k} + \frac{1}{\rho}\left(\frac{\partial P}{\partial s}\right)_\rho \Sigma\mathbf{k} + \Phi\mathbf{k} = 0, \tag{1.20}$$

$$k^2\Phi + 4\pi G D = 0, \tag{1.21}$$

$$\omega\Sigma = 0. \tag{1.22}$$

The plane wave solution and the equations 1.19-1.22, correspond to a variety of different physical situations. Taking $\omega = 0$ provides a solution with $\Sigma = 0$ and $\mathbf{k}.\mathbf{V} = \mathbf{0}$, which corresponds to vortical motion. However, of more interest are solutions where ω is non-zero and in particular solutions where $\mathbf{k}.\mathbf{V} = kV$. These solutions correspond to adiabatic sound waves provided $\Sigma = 0$.

Taking $\Sigma = 0$, it is seen that the solutions satisfy

$$\omega\delta_0 + kV = 0, \tag{1.23}$$

$$\omega V + k v_s^2 \delta_0 + k\Phi = 0, \tag{1.24}$$

$$k^2\Phi + 4\pi G\rho_0\delta_0 = 0, \tag{1.25}$$

where $\delta_0 = \frac{D}{\rho_0}$.

By eliminating $\Phi V \delta_0$ from these equations, it is found that the waves must obey a dispersion relationship,

$$\omega^2 - v_s^2 k^2 + 4\pi G\rho_0 = 0. \tag{1.26}$$

If ω is real oscillating waves occur, while if ω is complex then, depending upon its sign, growing and decaying perturbations are found. For ω to be complex it is necessary that,

$$v_s^2 k^2 - 4\pi G\rho_0 < 0, \tag{1.27}$$

which is equivalent to,

$$\lambda > \lambda_J, \tag{1.28}$$

where $\lambda_J = v_s \left(\frac{\pi}{G\rho_0}\right)^{1/2}$ is the 'Jeans' Length' and $\omega = \pm i(4\pi G\rho_0)^{(1/2)} \left(1 - \left(\frac{\lambda_J}{\lambda}\right)^2\right)^{1/2}$. The growing mode has an exponentially increasing amplitude with a characteristic time-scale $\tau \approx \frac{1}{|\omega|}$. Clearly at $\lambda = \lambda_J$ the time-scale is infinite, but if $\lambda \gg \lambda_J$ then $\tau \approx \tau_{ff} \approx (G\delta_0)^{-1/2}$ where τ_{ff} is the free fall gravitational collapse time.

The physics behind perturbation growth is very simple. The two competing effects in any perturbation are the self-gravity and the fluid pressure. As a crude approximation, it is reasonable to suggest that perturbations grow when their self-gravity overcomes the supporting fluid pressure. To get an estimate of when this occurs, the force per unit mass from self-gravity of a perturbation of size λ is,

$$F_g \approx G\rho\lambda, \tag{1.29}$$

and the force per unit mass due to pressure is,

$$F_p \approx \frac{v_s^2}{\lambda}. \tag{1.30}$$

Clearly $F_g > F_p$ if $\lambda > v_s^2 (G\rho)^{-1/2} \approx \lambda_J$.

There is an important caveat to this analysis that should be noted. For collisionless fluids the Euler equation is no longer valid and instead the Liouville equation, describing the evolution of the phase space distribution function, must be used in its place. Given this equation a perturbative approach can again be followed. Plane wave solutions are again found to exist, and gravitational collapse is prevented provided that $\lambda < \lambda_*$, where λ_* is the 'free-streaming length', defined by

$$\lambda_* = v_* \left(\frac{\pi}{G\rho_0}\right)^{-1/2}, \tag{1.31}$$

where

$$v_*^{-2} = < v^{-2} > = \frac{\int v^{-2} f d^3v}{\int f d^3v}, \tag{1.32}$$

corresponding to the mean of the velocity distribution for the particles.

The physics behind the prevention of growth of small-scale fluctuations is straightforward. Particles in too small a perturbation will free stream out of the perturbation (with velocity v_*) before the perturbation has a chance to grow. The limiting size of the perturbation necessary to allow collapse must then be proportional to v_*, which is clearly the case in 1.31. The effect of free streaming is also known as Landau damping or phase mixing.

1.3.2 Jeans' Instability Theory: Expanding Background Analysis

When modelling the physical universe the Hubble expansion must be taken in to account. The expansion serves to slow collapse, so that perturbations no longer grow or decay exponentially. The first analysis of a perturbation on an expanding background was done using General Relativity (Lifshitz, 1946), and the Newtonian analysis (Bonnor, 1956) did not follow until about 10 years later. Again, to keep the discussion simple, the Newtonian approximation is considered. The analysis of this situation proceeds in much the same way as the static case, although the time dependence is now incorporated in $D, \mathbf{V}, \Phi, \Sigma$, *i.e.* solutions are sought of the form,

$$\delta \begin{pmatrix} \rho \\ \mathbf{v} \\ \phi \\ s \end{pmatrix} = \begin{pmatrix} D(t) \\ \mathbf{V}(t) \\ \Phi(t) \\ \Sigma(t) \end{pmatrix} e^{i\mathbf{k}.\mathbf{r}}, \tag{1.33}$$

and the velocity about which perturbation occurs is $\mathbf{v} = \left(\frac{\dot{a}}{a}\right)\mathbf{r}$ and $\rho \propto a^{-3}$.

Substituting into equations 1.7-1.11, again yields conditions that $D, \mathbf{V}, \Phi, \Sigma$, must satisfy,

$$\dot{D} + 3\frac{\dot{a}}{a}D + i(\mathbf{v}.\mathbf{k})D + i\rho_1\mathbf{k}.\mathbf{V} = 0, \tag{1.34}$$

$$\dot{\mathbf{V}} + \frac{\dot{a}}{a}\mathbf{V} + i(\mathbf{v}.\mathbf{k})\mathbf{V} + iv_s^2\frac{D\mathbf{k}}{\rho_1} + i\left(\frac{\partial P}{\partial s}\right)_{\rho_1}\frac{\Sigma\mathbf{k}}{\rho} + i\Phi\mathbf{k} = 0, \tag{1.35}$$

$$k^2\Phi + 4\pi GD = 0, \tag{1.36}$$

$$\dot{\Sigma} + i(\mathbf{v}.\mathbf{k})\Sigma = 0, \tag{1.37}$$

as before v_s^2 is the adiabatic sound speed and ρ_1 is the unperturbed density. Once again longitudinal travelling wave type solutions, where $\Sigma = 0$ and $\mathbf{k}.\mathbf{V} = kV$, are chosen. Proceeding in a similar manner to the static background case it is found that,

$$\ddot{\delta\rho} + 2\left(\frac{\dot{a}}{a}\right)\dot{\delta\rho} + (v_s^2k^2 - 4\pi G\rho_1)\delta\rho = 0, \tag{1.38}$$

which is exactly analogous to the static case except that the $2\left(\frac{\dot{a}}{a}\right)\dot{\delta\rho}$ term acts as a frictional damping term, slowing the growth of perturbations. The growth of perturbations in different Universes can be examined. For example, for an Einstein-de Sitter universe, $a(t) \propto t^{2/3}$, $\rho = \frac{1}{6\pi Gt^2}$ and substituting these values into equation 1.38 yields,

$$\ddot{\delta} + \frac{4}{3}\frac{\dot{\delta}}{t} - \frac{2}{3t^2}\left(1 - \frac{v_s^2k^2}{4\pi G\rho}\right)\delta = 0, \tag{1.39}$$

which is seen to admit solutions of the form[2] $\delta = At^{2/3} + Bt^{-1}$, provided that $v_s^2k^2 \ll 4\pi G\rho$, or $k = 0$.

For completeness it should be mentioned how this analysis is performed for universes where the critical density[3], Ω is such that, $\Omega \neq 1$. In this case, the analysis has to be performed using the parametric variables associated with open and closed universes. For a full analysis see Weinberg (1972).

Analysing the development of perturbations in a radiation dominated universe is more difficult. Strictly speaking the Newtonian analysis cannot be used since radiation does not gravitate in the Newtonian picture. Nonetheless, a pseudo-relativistic analysis using the radiation equation of state $P = \frac{1}{3}\rho c^2$ can be performed. Homogenous solutions of the form $\delta = At + Bt^{-1}$ are found. This clearly corresponds to growing modes $\delta_+ \propto t$ and decaying modes $\delta_- \propto t^{-1}$.

Before concluding, it is worth mentioning another important type of perturbation, namely *isothermal* modes. These modes are not characterized by a fluctuation in the local space-time curvature, but instead correspond to fluctuations in the local equation of state. The simplest example of this would be a spatial fluctuation in the number of particles in a comoving volume, *i.e.* more baryons or dark matter particles. This variation in the equation of state is related to the fact that local pressure depends not only on pressure, but also upon composition (Kolb and Turner, 1990), *i.e.* the number of particles in a comoving volume.

[2]From now on the notation $\delta\rho = \delta$ will be used.

[3]Defined by $\Omega = 8\pi G\rho_0/3H_0^2$, where ρ_0 and H_0 are the density and Hubble constant for the universe at the t_0 epoch.

1.3.3 Zel'dovich Approximation

A key idea in the approximation is the difference between Eulerian and Lagrangian coordinate systems (Goldstein, 1980). In an expanding Universe, let $\mathbf{x}$ be a comoving coordinate, then it is related to the Eulerian physical coordinate using $\mathbf{r} = a(t)\mathbf{x}$. The comoving Lagrangian coordinate, $\mathbf{q}$, remains fixed in time and the dynamics of the system are encoded in the Lagrange map $\mathbf{s}(\mathbf{q}, t)$. The relationship between the two coordinate systems is given by,

$$\mathbf{r}(\mathbf{q}, t) = a(t)(\mathbf{q} - \mathbf{s}(\mathbf{q}, t)). \tag{1.40}$$

It should be clear that $\mathbf{s}$ acts as a displacement term, shifting mass away from the Hubble flow $a(t)\mathbf{q}$. To place constraints on $\mathbf{s}(\mathbf{q}, t)$ consider the effect of mass conservation;

$$\rho(\mathbf{r}, t)d^3r = \bar{\rho}\, d^3q \tag{1.41}$$

Rearrangement of the coordinate map between the systems shows that,

$$\rho(\mathbf{r}, t) = \bar{\rho}\det\left[\frac{\partial q_i}{\partial r_j}\right] = \frac{\bar{\rho}/a^3}{\det\left[\frac{\partial x_i}{\partial q_i}\right]}. \tag{1.42}$$

By writing $\mathbf{s}(\mathbf{q}, t) = f(t)\mathbf{p}(\mathbf{q})$, and $\rho_b(t) = \bar{\rho}/a^3$ this can be expressed as,

$$\rho(\mathbf{r}, t) = \frac{\rho_b(t)}{\det\left[\delta_{ij} + f(t)\partial p_j/\partial q_i\right]} \simeq \rho_b(t)(1 + f(t)\nabla_q.\mathbf{p}), \tag{1.43}$$

to first order in $f(t)$. Comparing this prediction for the dimensionless density contrast to the prediction from linear theory (section 1.3.2) gives

$$\frac{\delta\rho(\mathbf{r}, t)}{\rho_b(t))} = f(t)\nabla_q.\mathbf{p} = b(t)\delta(\mathbf{x}), \tag{1.44}$$

where $b(t)$ is the growing mode solution which, for example, is proportional to $t^{2/3}$ in a critical density Einstein-de Sitter universe. Given this result the identity, $f(t) \equiv b(t)$, is discovered and it is now possible to relate $\nabla_q.\mathbf{p}$ to the density contrast $\delta(\mathbf{x})$. Expanding $\delta(\mathbf{x})$ in Fourier modes at the initial time when $b(t) \simeq 0$ gives,

$$\nabla_q.\mathbf{p} = \sum_k A_k \exp(i\mathbf{k}.\mathbf{q}), \tag{1.45}$$

and hence,

$$\mathbf{p} = \sum_k \frac{i\mathbf{k}}{k^2} A_k \exp(i\mathbf{k}.\mathbf{q}). \tag{1.46}$$

$\mathbf{p}$ can then be represented by a potential Φ_0, and comparison to the linear perturbation theory shows that,

$$\Phi_0 = \frac{\phi}{3a\ddot{a}b} \tag{1.47}$$

where ϕ is the Newtonian potential corresponding to the perturbation.

In terms of this potential function, Φ_0, the peculiar velocity, $\mathbf{V}_p = \dot{\mathbf{r}} - H\mathbf{r}$, describing the perturbation from the Hubble flow, can be written,

$$\mathbf{V}_p = -a\dot{b}\nabla_q\Phi_0(\mathbf{q}). \tag{1.48}$$

and since $\mathbf{V}_p$ is the gradient of a potential it must be curl free, *i.e.* the velocity field is *irrotational*.

Returning to the density function $\rho(\mathbf{r}(\mathbf{q}), t)$, the determinant in equation 1.43 can be expressed in terms of the eigenvalues $\lambda(\mathbf{q})$ to give,

$$\rho(\mathbf{r}(\mathbf{q}), t) = \frac{\bar{\rho}/a^3(t)}{(1 + b(t)\lambda_1(\mathbf{q}))(1 + b(t)\lambda_2(\mathbf{q}))(1 + b(t)\lambda_3(\mathbf{q}))} = \bar{\rho}_b(t)\prod_{i=1}^{3}(1 + b(t)\lambda_i(\mathbf{q}))^{-1}. \tag{1.49}$$

This function becomes singular when $b(t) = -\frac{1}{\lambda_i}$. At this point *shell crossing* is said to have occurred and the 1-to-1 relationship between $\mathbf{q}$ and $\mathbf{r}$ is broken. In the physical universe shocks occur at this

point. By adding a viscosity term to the equation of motion, this problem can be removed (for a review see Shandarin and Zel'dovich, 1989).

Nonetheless, the density function still describes the onset of structure formation in a useful way. At early times when $b(t) \ll 1$ equation 1.49 can be approximated by,

$$\rho(\mathbf{q}, t) \approx \rho_b(t)\left(1 - b(t)(\lambda_1 + \lambda_2 + \lambda_3)\right). \tag{1.50}$$

Defining $\delta(t)$ by $\frac{\rho(\mathbf{q},t)-\rho_b}{\rho_b}$ then,

$$\delta(t) \approx -(\lambda_1 + \lambda_2 + \lambda_3)b(t). \tag{1.51}$$

Gravitational collapse is dictated by the eigenvalues λ_i and their associated eigenvectors. These are in turn governed by the over/under density on each eigenvector axis. In general, collapse occurs along a certain direction, although all possible collapse scenarios are summarized below,

1. $\lambda_1 \neq \lambda_2 \neq \lambda_3$ Collapse is the most common form of perturbation. In this case, one of the eigenvalues will be the most negative and collapse will occur along the direction of the associated eigenvector resulting in the formation of a caustic. The resulting structure is known as the 'Zel'dovich pancake'. Following this, collapse to a filament occurs as mass along the second eigenvector axis begins to collapse. The same occurs for the third axis, leading to the formation of a point-like object.

2. $\lambda_1 = \lambda_2 \neq \lambda_3$ (or any combination of two equal λ_i) Collapse occurs in two directions on approximately similar time-scales. The resulting structure is the filament mentioned as the second stage of collapse in (1). In much the same way as the first example, collapse may now proceed along the direction of the final eigenvalue.

3. $\lambda_1 = \lambda_2 = \lambda_3$ In this (rare) situation the tendency is to form the blobs mentioned earlier. However, such structures are expected to be rare (or at least on very large scales).

The Zel'dovich approximation is in essence a quasi-nonlinear solution (Bertschinger, 1992), because the linear trajectories of equation 1.40 are extrapolated beyond their truly applicable range. This linear displacement field is used to compute the density using equation 1.49, which is ***not a perturbation series.*** It is remarkable how much insight is gained from this procedure.

1.3.4 Relativistic Collapse Model

To calculate the collapse time of a perturbation it is useful to recall that, provided the region is spherical, Birkhoff's Theorem ensures that no external influence will be felt. For a spherical perturbation of constant density—a 'top hat'—the region will evolve in a exactly the same fashion as a closed universe. The equation of motion is trivially,

$$\frac{d^2R}{dt} = -\frac{GM}{R^2} \tag{1.52}$$

and the first integral,

$$\frac{1}{2}\left(\frac{dR}{dt}\right)^2 = E + \frac{GM}{R} \tag{1.53}$$

where E is the binding energy of the perturbation (and is thus required to be negative). Suppose that early on the collapsing system is moving almost in sync with the rest of the Hubble flow, *i.e.* peculiar velocities are negligible. Using an i subscript to denote this initial time the kinetic energy per unit mass of the outermost part of the sphere is,

$$K_i = \frac{1}{2}H_i^2 r_i^2 \tag{1.54}$$

while the potential energy is,

$$\frac{GM}{r_i} = \frac{4}{3}\pi G\rho_b(t)r_i^2(1+\delta_i) = K_i\Omega_i(1+\delta_i). \tag{1.55}$$

The total energy is then,

$$E = K_i\Omega_i[\Omega_i^{-1} - (1+\delta_i)], \tag{1.56}$$

and collapsing solutions ($E < 0$) require that $\delta_i < \Omega_i^{-1} - 1$.

At maximum expansion $\dot{r} = 0$, and hence

$$E = -\frac{GM}{r_m} = \frac{K_i \Omega_i (1+\delta_i) r_i}{r_m}, \tag{1.57}$$

relating this to the total energy derived in equation 1.56 gives an expression for the maximum radius (at t_m) parameterized by the initial values,

$$\frac{r_m}{r_i} = \frac{1+\delta_i}{\delta_i - (\Omega_i^{-1} - 1)}. \tag{1.58}$$

As would be expected, small perturbations take longer to collapse.

The general solution to equation 1.53 is given in parametric form by the evolution parameter θ, and corresponds to that of a cycloid. The radius and time evolution are given by

$$r(\theta) = A(1 - \cos\theta),$$
$$t(\theta) = B(\theta - \sin\theta),$$

where $A^3 = GMB^2$. After substituting, the constants A and B are found to be (for a critical density Einstein-de Sitter universe)

$$A = \frac{r_i}{2}\left(\frac{1+\delta_i}{\delta_i}\right)$$
$$B = \frac{1}{2H_i}\frac{1+\delta_i}{\delta_i^{3/2}}$$

Since for an E-dS universe $a \propto t^{2/3}$ and $\rho_b(t) = 1/(6\pi G t^2)$, the parametric equations can be used to write the density contrast as

$$\delta = \frac{9}{2}\frac{(\theta - \sin\theta)^2}{(1-\cos\theta)^3} - 1, \tag{1.59}$$

while linear theory predicts

$$\delta_L = \frac{3}{5}\delta_i \left(\frac{t}{t_i}\right)^{2/3} = \frac{3}{5}\left(\frac{3}{4}\right)^{2/3}(\theta - \sin\theta)^{2/3}. \tag{1.60}$$

In figure 1.1, the non-linear density contrast is compared to the linear estimate. The plot shows that the non-linear evolution quickly exceeds that predicted by linear theory, thus demonstrating that perturbation theory is comparatively unhelpful for studying the large density contrasts present in galaxies.

Although the non-linear solution (correctly) predicts collapse to an infinite mass point, random motions in the collapsing shells will lead to a system with asymmetry, which consequently virializes. By using the virial theorem, and the energy at the maximum expansion, it is found that the virialized sphere has a radius half that of the maximal expansion. From numerical experiments (see Coles and Lucchin, 1995), $t_{vir} \simeq 3t_m$, which combined with the results for the increased density suggests that the density contrast after virialization should be 400 times that of the background. This value is very close to that for rich clusters of galaxies. Note that at the collapse epoch the linear theory predicts a density contrast of; $\delta_{Lin}(t_c) \simeq 1.68$, far lower than the true value.

1.3.5 Statistics: Correlation Functions, Autocovariance and the Power Spectrum

This section provides a very brief overview of the statistical analysis applied to galaxy surveys. As a starting point for the analysis the assumption is made that the number of galaxies in a region is a direct indication of the over/under density of that region. Hence under this assumption *galaxies trace mass*. In cold dark matter models it is proposed that there is a *bias* parameter that affects how galaxies cluster relative to the underlying (and dominant) dark matter distribution. However, this idea is only relevant to how the data are interpreted in the final analysis and does not affect how it is formulated for analysis in the first place.

In the explanation of the Jeans' instability given in the previous sections, the idea of a single perturbation characterized by a wavevector $\mathbf{k}$ was utilized. In the physical universe there is a menagerie of perturbations, all associated with different wavevectors. So that progress can be made in the analysis, the following prescription for the density field is used,

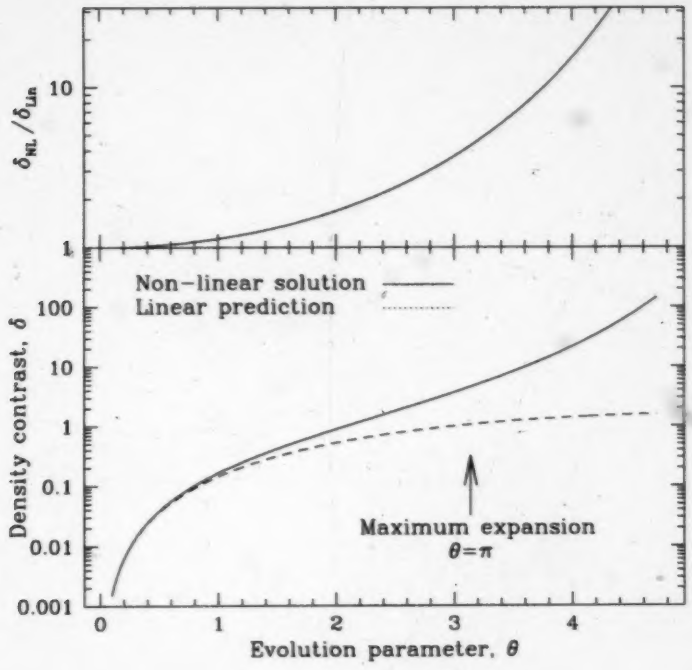

Figure 1.1: Comparison of the density contrast in the full non-linear solution to that predicted by linear perturbation theory. The ratio of the two is provided to show how quickly the linear theory fails to calculate the correct density contrast. At $\theta = \pi/2$ $\delta_{Lin} = 0.34$, while $\delta_{NL} = 0.47$ which is 40% higher. By $\theta = 2\pi/3$ the discrepancy has increased to almost 80%. By the maximum expansion epoch at $\theta = \pi$ the linear prediction is incorrect by approximately 300%.

1. $\rho(\mathbf{x},t) = \rho_1(1+\delta(\mathbf{x},t))$ where δ describes the density perturbation,
2. $\delta(\mathbf{x},t)$ is a Gaussian random field

Notably the idea that $\delta(\mathbf{x},t)$ be gaussian is supported by many calculations from inflation[4] (Starobinskii, 1982; Hawking, 1982; Guth and Pi, 1982).

The assumption that $\delta(\mathbf{x},t)$ is gaussian gives one immediate benefit,

- A complete statistical description of the spatial distribution of δ is provided by one function, the autocovariance function $\xi(r,t)$, and

$$\xi(r,t) = < \delta(\mathbf{x}_1,t)\delta(\mathbf{x}_2,t) > = \lim_{V\to\infty}\int_V \delta(\mathbf{x}-\mathbf{x}_1,t)\delta(\mathbf{x}-\mathbf{x}_2,t)d^3x, \tag{1.61}$$

where $r = \mid \mathbf{x}_1 - \mathbf{x}_2 \mid$ and the integral ranges over all spatial positions with fixed r. As alternative to taking the infinite limit, periodic boundary conditions over a finite volume can be imposed.

Since $\delta(\mathbf{x},t)$ is a combination of many different perturbations, the natural step to is to Fourier analyze,

$$\delta(\mathbf{x},t) = \frac{1}{(2\pi)^3}\int \delta(\mathbf{k},t)e^{i(\mathbf{k}.\mathbf{x})}d^3x, \tag{1.62}$$

and $\delta(\mathbf{k},t)$ may be written,

$$\delta(\mathbf{k},t) = \mid \delta_k \mid e^{i\phi}. \tag{1.63}$$

[4]Furthermore, any fluctuations away from a gaussian distribution may well be a signature of quantum gravity effects (Salopek, 1994).

Index	Name
n=0	Poisson fluctuations
n=1	Harrison - Zel'dovich spectrum
n=-3	Flicker-noise spectrum

Table 1.1: Names of common power spectrum indices.

This corresponds to requiring that each Fourier mode have a phase ϕ which is randomly distributed between 0 and 2π, and the distribution of amplitudes be given by a Rayleigh distribution. Equivalently, both the real and imaginary parts of $\delta(\mathbf{k},t)$ are normally distributed. Out of interest, note that the density field averaged over the largest scales in a simulation box cannot be Gaussian since there are too few modes contributing to the field. It is worth mentioning that $\xi(r,t)$ is the second moment, *i.e.* the variance, of the mass distribution and it can be shown to have no sensitivity to wall-like structures (Peebles, 1993). This is not significant since $\xi(r,t)$ averages out these effects in a similar way that gravity does, thus giving a reasonable description of large scale structure.

The Power Spectrum

Having arrived at the Fourier transform for $\delta(\mathbf{x},t)$ it is interesting to examine how $\xi(r,t)$ is described in k-space. To do this simply substitute 1.62 into the correlation function, *viz.*,

$$\xi(r,t) = < \delta(\mathbf{x}_1,t)\delta(\mathbf{x}_2,t) > = \frac{1}{(2\pi)^3}\int < \mid \delta_k \mid^2 > e^{i\mathbf{k}\cdot(\mathbf{x}_2-\mathbf{x}_1)}d^3k, \tag{1.64}$$

where $< \mid \delta_k \mid^2 >$ is called the *Power Spectrum* of the distribution. Clearly $< \mid \delta_k \mid^2 >$ is the Fourier transform of $\xi(r)$.

If it is assumed that mass fluctuations are isotropic, equation 1.64 can be simplified to,

$$\xi(r) = \frac{1}{(2\pi)^3}\int_0^\infty k^2 P(k)\frac{\sin kr}{kr}dk, \tag{1.65}$$

where P(k) now denotes the power spectrum.

In the description of the primordial power spectrum, the power law form, $P(k) \propto k^n$, where n is known as the power spectrum index, is often adopted. Some of the different indices are known by particular names, which are classified in Table 1.1. Early evolution of the power spectrum, during the radiation-dominated era, can be encapsulated in the *Transfer function* $T(k)$. Consequently, the initial power spectrum for galaxy formation studies is of the form $P(k) = k^n T(k)$.

There are some problems interpreting the power spectrum as it stands. It is quite possible that it may be influenced by small scale fluctuations that are not of interest. This problem can be alleviated by using a filter/smoothing function, W, such that,

$$\bar{\delta}(\mathbf{x}) = \frac{1}{V}\int \delta(\mathbf{x}')W(\mid \mathbf{x}-\mathbf{x}' \mid)d^3x', \tag{1.66}$$

where $\bar{\delta}(\mathbf{x})$ is now a smoothed density function. W may be a gaussian filter (see below) or it may be a sharp filter depending upon the data and the analysis desired. One example of a filter function is the Gaussian smoothing 'window', $W_g(\mathbf{x})$, where,

$$W_g(\mathbf{x}) \propto e^{-\frac{x^2}{2x_g^2}}, \tag{1.67}$$

and x_g corresponds to the characteristic length that the function will smooth over. See Peebles (1993) for a discussion of different types of filter functions and their applications. The analysis of $\bar{\delta}$ is no more difficult than that of δ, the correlation function is given by,

$$< \bar{\delta}^2(r) > = \Sigma^2(r) = \frac{1}{(2\pi)^2}\int k^2 P(k)\tilde{W}^2(kr)dk, \tag{1.68}$$

where $\tilde{W}$ is the Fourier transform of W.

1.4 Galaxy formation via analytic and semi-analytic methods

1.4.1 An analytic model of galaxy formation

Since galaxy formation is a highly asymmetric process, analytic models, which consider uniform or 1-dimensional systems, are somewhat limited. Further, a perturbative analysis is not valid since galaxies are formed in the highly non-linear regime. Nonetheless, this has not discouraged theorists from developing a quantitative model of galaxy formation within dark matter halos (White and Rees, 1978; Fall and Efstathiou, 1980). This section draws heavily from the review by White (1994).

Uniform cloud model

Galaxy formation occurs as baryons accumulate within dark matter halos. Thus, the model should include both gas dynamics and gravity. Radiative cooling (bremsstrahlung) is the third part of the model since it is necessary to be able to cool the gas. Otherwise the baryons would collapse down to a size where the gas pressure provides support against self-gravity and the external potential from the dark matter. If the gas mass of the system is a fraction of the total mass, say $M_g = fM$, then the pressure force per unit mass of gas remains v_s^2/λ, while the gravitational force becomes $G(\rho\lambda^3)^2/\lambda^2/(\rho_g\lambda^3) = G\rho\lambda/f$. Equating the pressure and gravitational forces gives $\lambda = f^{1/2}\lambda_j$, where $\lambda_j = v_s(G\rho)^{-1/2}$ is approximately the Jeans' length of a gaseous self-gravitating system. Thus as the fraction of gas becomes smaller so does the Jeans' length, as would be expected.

At sufficiently high temperatures (10^6 K) the primordial gas (assumed to be solely hydrogen for the purposes of this explanation) will be fully ionized. The power output due to radiative cooling per unit volume is given by,

$$\frac{dE}{dt} \propto n_e n_H T^{1/2} \tag{1.69}$$

where n_e is the number of electrons, n_H the number of hydrogen ions and T is the temperature of the gas. The true picture is more complex than this since primordial gas is a mixture of H and He (76% *vs.* 24% by mass) and the gas is rarely above the limit 10^6 K. Below 10^6 K the physics is significantly more complex with emission coming from different transitions (*e.g.* H, He recombination, partial ionization). However, the cooling curve can still be parameterized as a function of the number of electrons and ions, which in turn can be written in terms of the gas density. Thus the power output from radiative cooling may be summarized,

$$\frac{dE}{dt} = \rho^2 \Lambda(T) \tag{1.70}$$

where $\Lambda(T)$ is known as the "cooling curve". It is actually comparatively difficult to calculate since the exact ionization fraction of each species must be known for any given temperature. A plot of the cooling curve for a primordial admixture of He and H from Sutherland and Dopita (1993) is given in figure 1.2.

The largest object that can form a galaxy will be one for which the cooling time is approximately equal to the collapse time. Thus an estimate for the cooling time of a gas cloud must be found. Assuming that the gas cloud has had sufficient time to evolve it will be in acoustic oscillation at the Jeans' Length, with a mass $\simeq \rho\lambda^3$. This result can also be derived by using the virial theorem, *viz.*,

$$2K + W = 0, \tag{1.71}$$

giving,

$$2\frac{3}{2}\frac{kT}{\mu m_p} = \frac{3}{5}\frac{GM}{R} = \frac{3}{5}\frac{GM_g}{fR}, \tag{1.72}$$

where μ is the relative molecular mass. The LHS is the gas kinetic energy (multiplied by the virial prefactor) whilst the RHS is the potential energy of a uniform sphere of radius R and mass M. Rearranging for M_g yields

$$M_g = \frac{5fkT}{(G\mu m_p/R)}, \tag{1.73}$$

which indicates that the Jeans' Mass is (approximately) set by the ratio of the thermal energy of an atom to the potential at a distance R. After substituting for R by calculating the gas mass in a sphere of radius R with gas particle number density n, it is found that,

$$M_g = \left(\frac{5k}{G\mu m_p}\right)^{3/2} \frac{1}{(4\pi\mu m_p/3)^{1/2}} f^{3/2} T^{3/2} n^{-1/2}. \tag{1.74}$$

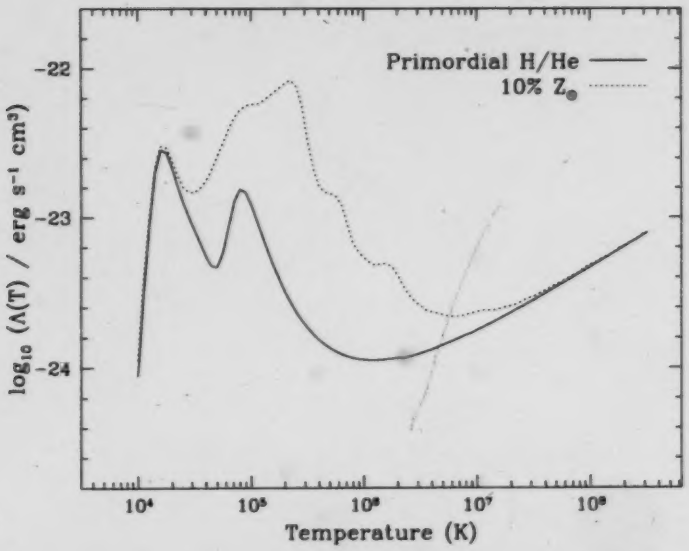

Figure 1.2: Plot of the cooling function $\Lambda(T)$ normalized so that $\Lambda(T)n^2$, where n is the total number density of ions, is the power output per unit volume. Curves for both the primordial composition (76% H, 24% He) and a composition with 10% solar metallicity ($Z_\odot$) are shown. Adding metals to the plasma increases the cooling rate at 2×10^5 K by an order of magnitude. The high temperature free-free emission is comparatively unaffected by the addition of metals. The steep slopes in this log-log plot indicate that the rate of change of cooling with respect to temperature can be very fast, making accurate numerical integration difficult.

Using a parameterization suggested by White,

$$M_g = 7.9 \times 10^{12}\, T_6^{3/2}\, f^{3/2}\, n_{-3}^{-1/3}\, \mathrm{M}_\odot \tag{1.75}$$

where $T = 10^6\ T_6$ K, $n = 10^{-3}\ n_{-3}$ cm^{-3}. As expected from the Jeans' Mass argument, an expression dependent upon $f^{3/2}$ has been derived. Note this value is lower than that in White (1993) since $\mu = 0.6$ has been used rather than $\mu = 0.5$.

This value may be rewritten in terms of the collapse mass at a particular redshift. Since density scales as the inverse cube of the expansion parameter,

$$n_{-3} = n^0_{-3}(1+z)^3. \tag{1.76}$$

Scaling to the over-density of the collapsing cloud relative to the background cosmological density,

$$n_{-3} = \frac{\rho}{\bar{\rho}}\,\frac{f\bar{\rho}^0_{-3}}{\mu m_p}\,(1+z)^3. \tag{1.77}$$

and writing this in terms of the over-density $\delta = (\rho/\bar{\rho} - 1)$, the Hubble parameter $H_0 = 100h$ km s^{-1} Mpc^{-1} and the density parameter $\Omega_0 = (8\pi G\rho_0/3H_0^2)$, yields,

$$n_{-3} = 1.9 \times 10^{-2}\, f(1+\delta)\,(\Omega_0 h^2)\,(1+z)^3. \tag{1.78}$$

Substituting this result into the Jeans' Mass derived in equation 1.75, gives,

$$M_g = 5.4 \times 10^{13} T_6 f(1+\delta)^{-1/2}\,(\Omega_0 h^2)^{-1/2}(1+z)^{-3/2}\mathrm{M}_\odot \tag{1.79}$$

which is the Jeans' Mass parameterized in terms of the over-density of an object at a given redshift. Given this parameterization the virial temperature resulting from a given overdensity at a certain redshift can be found. For example, for an overdensity $\delta \sim 200$ in a $\Omega_0 h^2 = 0.25$ universe, a collapse mass of 10^{11} $M_\odot$ at $z = 3$ leads to a virial temperature of $T = 2.1 \times 10^6$ K.

The next question to ask is: *given this system at the temperature found, is the gas cooling time sufficiently short to allow the system to collapse further?* If the cooling time of the gas is shorter than the free fall time, then heat can be radiated away sufficiently fast to prevent pressure support occurring.

To a reasonable approximation, since the cooling function is monotonically increasing above $T = 10^5$ K, the cooling time of the gas can be written

$$t_{cool} \simeq \frac{E_{gas}}{dE/dT} = \frac{\frac{3}{2}nkT}{n^2\Lambda_{min}}, \tag{1.80}$$

where Λ_{min} is the minimum of the cooling function, roughly $\sim 10^{-24}$ erg cm^3 s^{-1}. The free fall time is approximately

$$t_{coll} \simeq \pi\sqrt{\frac{R^3}{GM}} = \left(\frac{3\pi f}{4Gn\mu}\right)^{1/2}, \tag{1.81}$$

and equating the two yields the limit condition:

$$\sqrt{\frac{\pi}{3k^2G\mu}} f^{1/2} n^{1/2} \simeq \frac{T}{\Lambda_{min}}, \tag{1.82}$$

which divides collapsing objects into those that can undergo unimpeded collapse and those which reach pressure support and cannot collapse further (the Rees-Ostriker Criterion, Rees and Ostriker, 1977). This division is extremely important since it places an upper limit on the mass of objects that can collapse to form galaxies. Above 10^5 K the cooling function obeys the relation, $\Lambda \propto T^{-1/2}$, which after appropriate normalization can be used to rewrite equation 1.82 as,

$$T_6^{3/2} f^{-1/2} n_{-3}^{-1/2} \simeq 2.5, \tag{1.83}$$

and after substituting into the Jeans' Mass equation 1.75, the limiting mass is found to be,

$$M_g \simeq 2 \times 10^{13} f^2 \mathrm{M}_\odot = f^2 M_{limit}. \tag{1.84}$$

Thus condensation is not possible within a halo of total mass M_{limit}. The quadratic dependence on f shows that an $f = 1$ cosmology will have a markedly different maximum collapse mass (2×10^{13} $M_\odot$) compared to an $f = 0.1$ cosmology (2×10^{11} $M_\odot$). Table 1.2 compares collapse times and cooling times for a number of different overdensities and cosmologies.

To understand these concepts in more detail, it is helpful to resort to a phase space plot of the collapse mass versus redshift and the associated cooling time of the object. See figure 1.3 for the plot and its interpretation. The fundamental point of this analysis can be summarized thus: the maximum mass of galaxies is set by the limit from the Rees-Ostriker (cooling) criterion.

Torques and the growth of spin

The growth of the angular momentum of protogalaxies is a problem ideally suited to the Zel'dovich approximation. By extending the trajectories into the quasi-non-linear regime, predictions can be made to a later epoch than that possible by Eulerian techniques.

To see how the analysis unfolds, assume that a perturbation occupies a region V. The angular momentum about the barycenter $\bar{\mathbf{r}} = \int_V \mathbf{r} dV/V$ is then,

$$\mathbf{J} = \int_V \rho_b(t)(\mathbf{r} - \bar{\mathbf{r}}) \times (\mathbf{v} - \bar{\mathbf{v}})\, dV. \tag{1.85}$$

Converting this to the Lagrangian volume integral gives,

$$\mathbf{J}(t) = \bar{\rho} a^5 \int_{V_L} (\mathbf{q} - \bar{\mathbf{q}} - [\mathbf{s}(\mathbf{q},t) - \mathbf{s}(\bar{\mathbf{q}},t)]) \times -(\dot{\mathbf{s}}(\mathbf{q},t) - \dot{\bar{\mathbf{s}}}(\bar{\mathbf{q}},t))\, d^3q, \tag{1.86}$$

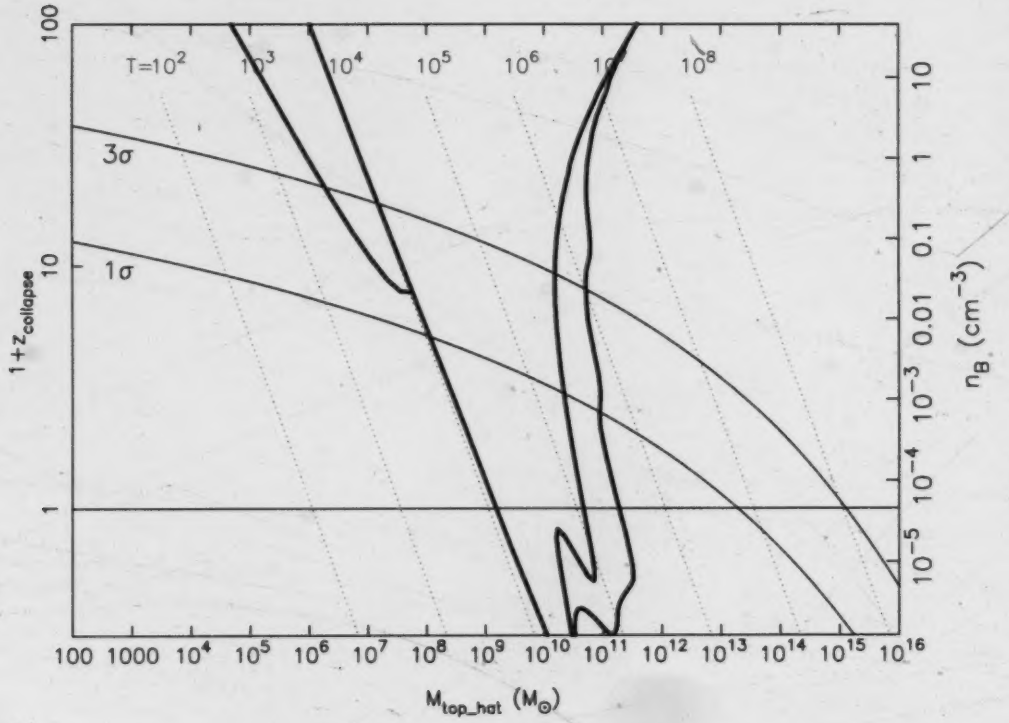

Figure 1.3: Plot of the parameter space of collapsing perturbations, including the effect of gas cooling. The 3σ and 1σ lines correspond to spherical overdensities measured relative to the mass variance calculated for a power spectrum normalized to reproduce the mass variance of observed clusters. The bottom axis gives the mass present in the overdensity, the left axis corresponds to the collapse epoch and the right hand axis gives the density of the object, inferred by inverting the free fall collapse time equation ($\rho \propto t^6/M^2$). The horizontal line at $(1+z) = 1$ corresponds to objects that have collapsed at the current epoch. The dotted lines are lines of constant Jeans' Mass in a density-temperature plane inferred from the density values on the right hand axis. The temperature values at the end of each line are marked, with a spacing of one decade in temperature. Since constant M_J leads to the relation $T \propto n^{1/3}$ the lines have a slope of 1/3. The dark lines are relevant to the density-temperature plane, and delineate the point where the ratio of the cooling time to collapse time is unity, *i.e.* they denote the Rees-Ostriker criterion. The left-most line corresponds to the cooling curve for primordial gas while the right for 5% solar metallicity. Starting from the right hand side of the plot the overdensity lines correspond to regions where the cooling time exceeds the free fall time. In this region pressure support is possible and unimpeded collapse does not occur. Moving leftward along the overdensity lines eventually the Rees-Ostriker line is reached and to the left of this line objects can cool within the collapse time. Hence there is no pressure support, *i.e.* unimpeded collapse occurs. Moving further left hits the other side of the cooling criterion where objects can no longer cool sufficiently quickly. The second dark line on the left hand side corresponds to the cooling from molecular hydrogen, relevant at low temperature. Clearly there is a region between $10^9 - 10^{11}$ $M_\odot$ where collapse can occur unimpeded.

z	δ	$\Omega_0 h^2$	f	M_g (10^{11} $M_\odot$)	T (10^6 K)	n_{-3} (10^{-3} cm^{-3})	t_{cool} (yr)	t_{coll} (yr)
3	200	0.25	0.05	1.0	2.10	3.06	3.92×10^9	9.82×10^8
1	200	0.25	0.05	1.0	0.74	0.38	1.11×10^{10}	2.78×10^9
5	200	0.25	0.05	1.0	3.86	10.3	1.87×10^9	5.34×10^8
3	200	0.25	0.05	0.10	0.21	3.06	1.66×10^8	9.82×10^8
3	200	0.25	0.05	10.0	21.0	3.06	1.87×10^{10}	9.82×10^8
3	200	0.25	0.05	100.	210.	3.06	6.74×10^{10}	9.82×10^8
3	200	0.25	0.10	1.0	1.05	6.11	1.01×10^9	9.82×10^8
3	200	0.25	1.0	1.0	0.11	61.1	1.19×10^6	9.82×10^8

Table 1.2: Cooling times and collapse times for primordial clouds characterized by redshift, over-density, $\Omega_0 h^2$ and baryon fraction. The first line gives the cooling and collapse times for a protogalaxy at z=3 parameterized by fiducial values for the over-density, $\Omega_0 h^2$ and baryon fraction. The first grouping of data demonstrates that at $z = 5$ a cloud of over-density of 200 is a significantly denser object than one at $z = 3$, and conversely for $z = 1$. This leads to both shorter collapse and cooling times at higher z, while longer times at low z. The second grouping compares the effect of changing the mass of the cloud (assuming constant density). For higher masses the cooling time is increased since the virial temperature becomes higher. Consequently the cooling time is significantly longer than the collapse time (as is observed in clusters of galaxies). At lower masses the cooling time is shorter than the dynamical time, thus in a hierarchical cosmology the gas in progenitor halos is expected to cool. The final grouping compares the effect of varying the baryon fraction. As the baryon fraction is increased the virial temperature drops, the gas density increases and the cooling time falls very rapidly.

Since the cross product of $\mathbf{s}(\mathbf{q},t) - \mathbf{s}(\bar{\mathbf{q}},t)$ with its time derivative vanishes, the above equation reduces to,

$$\mathbf{J}(t) = \bar{\rho} a^5 \int_{V_L} (\mathbf{q} - \bar{\mathbf{q}}) \times -(\dot{\mathbf{s}}(\mathbf{q},t) - \dot{\bar{\mathbf{s}}}(\bar{\mathbf{q}},t))\, d^3q. \tag{1.87}$$

At this point the Zel'dovich approximation $\mathbf{s}(\mathbf{q},t) = b(t)\nabla_q \Phi_0(\mathbf{q})$ is applied to give,

$$\mathbf{J} = \bar{\rho} a^5 \dot{b} \int_{V_L} (\mathbf{q} - \bar{\mathbf{q}}) \times \nabla \Phi_0\, d^q q, \tag{1.88}$$

and thus $\dot{b}(t)$ parameterizes the temporal evolution of the angular momentum in concert with the expansion parameter $a(t)$.

Further progress can be made by expanding the gradient of the potential about the barycenter $\bar{\mathbf{q}}$ (assuming the potential is smooth enough to do this). The i-th component is given by,

$$\frac{\partial \Phi_0}{\partial q^i}(\mathbf{q}) = \frac{\partial \Phi_0}{\partial q^i}(\bar{\mathbf{q}}) + (q^j - \bar{q}^j) \left[\frac{\partial^2 \Phi_0}{\partial q^j \partial q^i} \right]_{\bar{\mathbf{q}}} \tag{1.89}$$

Since the integration is performed about the barycenter, the integration over the first term of the expansion is zero. Thus there is no overall angular momentum from integrating the momentum associated with the potential gradient at $\bar{\mathbf{q}}$ around the barycenter (as would be expected).

Writing out the j-th component of the remaining terms yields,

$$J_i(t) = -a^2 \dot{b} \epsilon_{ijk} \left[\frac{\partial^2 \Phi_0}{\partial q^j \partial q^i} \right]_{\bar{\mathbf{q}}} \int_{V_L} (q^j - \bar{q}^j)(q^k - \bar{q}^k) \rho a^3 d^3q \tag{1.90}$$

Note that the integral is the inertia tensor, $\mathcal{I}$ for the perturbation, while the potential term is the tidal tensor, $\mathcal{T}$. This result can be rewritten using index notation for $\mathcal{I}$ and $\mathcal{T}$,

$$J_i(t) = -a^2 \dot{b} \epsilon_{ijk} \mathcal{T}_{jl} \mathcal{I}_{lk}. \tag{1.91}$$

Consequently a direct estimate of the growth of the angular momentum can be made. In an Einstein-de Sitter universe $a(t) \propto t^{2/3}$, $\dot{b}(t) \propto t^{-1/3}$, giving,

$$J_i(t) \propto t, \tag{1.92}$$

i.e. the angular momentum grows linearly with time, which is confirmed by simulation (White, 1984).

The principal axes of $\mathcal{T}_{jl}$ and $\mathcal{I}_{lk}$ do not, in general, coincide as can be seen by writing out a component of J_i, *viz.*,

$$J_1 \propto \mathcal{T}_{23}(\mathcal{I}_{33} - \mathcal{I}_{22}) + \mathcal{I}_{23}(\mathcal{T}_{22} - \mathcal{T}_{33}) + \mathcal{T}_{21}\mathcal{I}_{13} - \mathcal{T}_{31}\mathcal{I}_{12}, \tag{1.93}$$

which in a principal-axes representation for $\mathcal{I}$ ($\mathcal{I}_{23} = \mathcal{I}_{13} = 0$) gives

$$J_1 \propto \mathcal{T}_{23}(\mathcal{I}_{33} - \mathcal{I}_{22}) \tag{1.94}$$

Thus for J_1 to be non-zero it is necessary that $\mathcal{T}_{23}$ be non-zero, and therefore the principal axes of $\mathcal{T}$ cannot coincide with those of $\mathcal{I}$.

Equation 1.91, demonstrates that the origin of spin is due to the coupling of the inertia tensor to the tidal tensor. In particular it is the quadrupole moment generated by the irregular boundary of V_L that couples to the tidal field. Notably, if V_L is spherical then there is no angular momentum[5]. This can be seen explicitly by using Gauss's Theorem to rewrite equation 1.88 as a surface integral,

$$\mathbf{J}(t) = -\bar{\rho}a^5\dot{b}\int_{\Sigma_L} \Phi_0(\mathbf{q})(\mathbf{q} - \bar{\mathbf{q}}) \times d\mathbf{S} \tag{1.95}$$

thus $\mathbf{J}$ vanishes if V_L is spherical or Σ_L is an equipotential of Φ_0.

1.4.2 Semi-analytic models of galaxy populations

Strictly speaking, the semi-analytic approach (Kauffmann *et al.*, 1993; Cole *et al.*, 1994; Somerville and Primack, 1998) is not a theory of galaxy formation but rather one of galaxy populations. The approach has a somewhat 'cookbook' feel to it since a number of parameters must be constrained from simulations. The foundation of semi-analytic models is the "extended Press-Schecter theory" (Bond *et al.*, 1991) which can be used to describe the formation of dark matter halos and in particular the merger rates between them. Note that for accurate calculations the theory requires calibration from N-body simulations (Lacey and Cole, 1994).

To realize a galaxy population, first the desired cosmology must be chosen by specifying the density parameter, Hubble constant, cosmological constant, baryon fraction, and finally the spectrum of initial perturbations. Once this has been done, many different galaxy size halos can be constructed using Press-Schechter theory from a series of random walks. An ensemble average over all the halos can then be used to construct properties of the expected galaxy population as a whole. The detailed physics of the baryons (*i.e.* star formation, feedback from supernovae, relative amounts of cold and hot gas and metal content) are calculated for each of the progenitor halos in a galaxy size halo. Consequently the evolution of the galaxy size halo, determined by the 'merger tree', can be incorporated into the physics of the progenitor halos. For example, if two progenitor halos merge then the result is a larger halo with a higher density. If the metal content of the progenitor halos are different then the new halo will have a different average metallicity compared to the progenitors. Clearly, a number of simplifying assumptions have to be made about the merger process, which is one of the most unsatisfying parts of the theory. White (1994) summarizes the stages in constructing a semi-analytic model as follows:

1. Select the cosmology and specify its parameters.
2. Construct a number of different galaxy halos with different merger trees.
3. For each progenitor halo assign characteristics associated with the baryonic component: (a) fraction of hot gas (*i.e.* that which is above 10^5 K and consequently X-ray emitting), (b) fraction of cold gas and (c) fraction of stars.
4. Choose a star formation efficiency and make assumptions about feedback from supernovae, such as metal enrichment and the fraction of cold gas converted to hot.

[5]This result is a consequence of the first order approximation. White (1984) shows that at second order a sphere does develop angular momentum.

5. Apply physics to the evolution of the system. For example, star formation converts cold gas to stars, supernovae feedback converts cold gas to hot and radiative cooling converts hot gas to cool.

6. Use a stellar population model (*e.g.* Bruzual and Charlot, 1993), to monitor the evolution of the stellar colours and luminosity,

7. Make simplifying assumptions about the physics of mergers, especially what happens to the baryonic components. This covers the effect of mergers on disc disruption and the morphology of objects.

8. Once all the halos have been realized, perform the ensemble averaging to get the results for a particular cosmology.

The benefits of the semi-analytic approach are that it allows a wide number of halos to be examined very quickly, far more so than can be achieved in simulations. The biggest flaw in the method is that it requires a number of assumptions to be made about the physics of mergers, some of which may well be incorrect. Recently, researchers have combined collisionless n-body simulations with semi-analytic methods to place galaxies 'by hand'. The advantage over the standard semi-analytic method is that the merger tree is calculated exactly; however the assumptions about the evolution of the baryonic component remain.

1.5 The utility of simulation

Numerical models provide insight when the underlying theoretical model is too complex to be solved analytically. Typically, this occurs when the phenomenon being modelled is highly non-linear, geometrically complex or, alternatively, includes a large number of physical processes. In this sense simulation can be viewed as very much an experimental science, whose main benefit is to bridge the gap between knowledge of the physics of a phenomenon and a theory of the phenomenon itself (which uses a number of simplifications). Without an understanding of the underlying physics there can be no simulation. Further, without analytic models to refer to, the accuracy of simulations cannot be verified.

In the event that a number of experimental results, or observations, are available, simulation can be used in a rigorous fashion. For example, if a simulation does not provide a reasonable reproduction of (trusted) observations it is apparent that significant physics is missing from the model even before a theory of the process under study can be formulated (provided of course that the simulation algorithm has been tested sufficiently to ensure its accuracy). Thus in this scenario the theory must be revised, which in turn will lead to a revision of the simulation algorithm. This period of "postdiction" is essential to progress. Norman (1996) points out that the development of simulation and theory advance along similar lines. In the development of any numerical model the following three considerations must be made,

1. Dimensionality. Early work in a field often uses low dimensionality, to render the problem tractable with the computational power available.

2. Resolution. Increased resolution can often be used to gain insight into a problem before increased dimensionality.

3. Physical complexity. In mature fields, further progress is often made by including additional physics rather than adding extra resolution.

Considerations 1 & 3 clearly apply to any analytic model as well.

Once a numerical model has been shown to be valid, either by reproduction of analytic results or observations, significant progress can be made. Simulations provide more information than observations alone, especially temporally, as the time evolution of the system under study can be elucidated. The ability to examine different physical variables provided by a simulation is also especially beneficial. I denote simulations used in this fashion as "interpolative" simulations. Such simulations provide the insight necessary to form a simplified theory of the phenomenon under scrutiny. Large-scale structure simulations can be viewed as interpolative simulations, especially in view of lack of physical complexity. The initial conditions are comparatively well known (from the COBE data) and so are the final (from redshift surveys).

The most powerful, and potentially misleading, mode of use for simulations is the "extrapolative" mode. In the situation where observations or experimental results are not available, simulations can provide a numerical laboratory. Of course, such simulations are dangerous in that they may become misleading when new parameter spaces are explored which require a new underlying theory. Thus,

for this mode of use, part of the 'art' is understanding the physics that is left out of a simulation. The Accelerated Strategic Computing Initiative (ASCI, U. S. Department of Energy, 1996) programme is an entirely extrapolative one, as the whole point of it is to avoid the requirement of experimental results (*i.e.* nuclear testing).

Simulations of galaxy formation sit on the border of the extrapolative and interpolative modes. They are interpolative in the sense that investigators are trying to reproduce a wealth of observations available from the local Universe. However, they are also extrapolative since very little information is available about the high redshift objects that merge to form galaxies, which is ***fundamental*** in the construction of the numerical model. This lack of observations manifests itself in the underlying theory as a poor understanding of how to model star formation and feedback. In this thesis it will be argued that simulation of galaxy formation is difficult because, at least for the moment, it is an extrapolative endeavour. Fortunately, advances in spectroscopy and larger telescopes will serve to reduce this problem. It should be noted that even after the MAP and Planck Surveyor satellites have mapped the initial conditions in minute detail, galaxy formation will still be a flourishing field.

1.6 Particle simulation methods

Computer simulations using particles can be applied to a wide variety of fields, from plasma physics to cosmology. A short summary of a number of important algorithms is given (see Hockney and Eastwood, 1988 for an extensive review, from which this discussion is based). For the purposes of this section it will be assumed that the force being modelled is gravitational.

1.6.1 Particle-Particle method

The conceptually simplest method for calculating the evolution of n-bodies interacting is to perform a pair-wise summation for each particle with the remaining particles in the simulation—the Particle-Particle (PP) method. Thus to recover the force on particle i a summation of all the forces from other particles must be performed, *viz.*,

$$\mathbf{F}_i = \sum_j \mathbf{F}_{ij}. \tag{1.96}$$

For a simulation with n-bodies, approximately αn^2 where $\alpha \simeq 10$, operations are required to advance the particles one time-step. This is computationally efficient for $v \leq 1000$ but because of the n^2 term it rapidly becomes very expensive. Since in a simulation some particles will be travelling faster than others, some particles require a shorter time-step for accurate integration of the orbit than others. If the algorithm can be written so as to allow different particles to be updated with different time intervals, then a significant amount of work can be saved since it is not necessary to update all the particles using the shortest time interval. This 'multiple time-step' approach can be applied to the PP method and saves a significant amount of time in simulations where binary systems form (particularly simulations of star clusters).

Because of the simplicity of this algorithm it was the first to be widely used within astrophysics (Hoerner, 1960; Aarseth, 1963; Henon, 1964). Aarseth published a seminal series of papers (Aarseth, 1963; Aarseth, 1966; Aarseth, 1969) that examined the evolution of a galaxy cluster with 100 galaxies. During the 1970's this work was extended to simulations with up to 4,000 galaxies (Groth *et al.*, 1977; Aarseth *et al.*, 1979). In related work the PP method was first applied to star clusters by Aarseth (1971). Because of the high force accuracy provided by the PP method it is particular important in simulations where strong 2-body interactions occur (globular clusters for example). Recently, speed up in the algorithm has been achieved by creating hardware circuits with the calculation encoded in them—the GRAPE series of computers (Sugimoto *et al.*, 1990). This solution has allowed the number of particles to be increased to 500,000, but the new hardware remains handicapped by the n^2 scaling.

1.6.2 Particle-Mesh method

The basis of the Particle-Mesh (PM) method is that a Lagrangian particle representation of the data may be mapped to an Eulerian grid representation using an interpolation function. After the data has been mapped to a grid, a rapid solution method can be used to find the solution to the field equation on the grid (for gravity the solution to Poisson's equation must be found). Once the solution for the force is found on the grid it must then be transposed back to the particles, which is essentially the inverse of the interpolation operation used to assign the particle data to the grid.

The interpolation process reduces the accuracy of the method, since the resolution becomes dependent upon the fineness of the grid to which interpolation occurs. However, provided the interpolation, solution and de-interpolation step are fast enough (*i.e.* have a functional dependence that is not order n^2) then the algorithm will be faster than the PP algorithm (once a sufficiently large n is reached, typically 1,000). The interpolation and de-interpolation step are order n and the solution step is order $\beta(n)$ (*i.e.* $5n \log_2 n$ for a mesh of size $L = n^{1/3}$ using a Fast Fourier Transform) so the operation count is,

$$\text{Operations} = \alpha n + \beta(n) \simeq 20n + 5n \log_2 n, \tag{1.97}$$

and a simple calculation shows that for 32^3 particles this method is about 20,000 times faster than the PP calculation.

The PM method was developed by plasma physicists who were quick to see the shortcomings of the PP algorithm as applied to systems with large numbers of particles. Seminal work in 1-dimension was conducted by Burger (1965) and in 2-dimensions by Hockney (1966). The most significant development in the PM algorithm has been the optimization of the force calculation to remove grid 'noise' inherent in the interpolation and solution steps. Such 'Quiet Particle-Mesh' codes (see, Hockney and Eastwood, 1988) have a noise level hundreds of times lower than the early simulations.

In astrophysics, the PM method was introduced by Hockney (1967), who performed 2-dimensional simulations of galaxy evolution using a 2-dimensional force law. The PM method is suited to this kind of calculation since the galaxy is modeled using 'superparticles' containing hundreds of thousands of stars which may pass through each other—a process mimicked by the finite grid resolution. Hohl and Hockney (1969) performed calculations of a 3-dimensional force law with particles constrained to move in 2-dimensions. The high efficiency of the PM method coupled with the 2-dimensional simulation allowed simulations with 200,000 particles to be conducted, and effective investigations of spiral structure to be performed (Hohl, 1971). 3-dimensional simulations of galaxy evolution were first performed by Hockney and Brownrigg (1974). The PM method is also widely used in calculations of galaxy clustering in expanding universes, although the lack of sub-grid resolution is undesirable.

1.6.3 Particle-Particle, Particle-Mesh method

The motivation in the development of the Particle-Particle, Particle-Mesh (P^3M) method is to overcome the short-range limitation of the PM method, by resolving forces at length scales smaller than the grid spacing. The force is thus decomposed into a short-range component (found by evaluating a PP calculation over *local* particles) and a long range component found from a PM calculation. As for the PM method, P^3M was initially developed for electrostatic plasma calculations.

Since P^3M incorporates both a PM calculation and a short range PP calculation the operation count is,

$$\text{Operations} = \alpha n + \beta(n) + \gamma n_{short} n. \tag{1.98}$$

n_{short} is the average number of short range neighbours, given approximately by $4\pi a^3 \bar{n}/3$ where a is a grid spacing set by the force accuracy required and $\bar{n}$ is the average particle density. By a suitable choice of n and a the algorithm can be made to scale almost linearly in n. Note, however, that in the gravitational calculation n_{short} may become very large leading to significant slowdown in the calculation. Thus it is always beneficial to use as large a mesh as possible for this type of problem. In later chapters an alternative solution by Couchman (1991) will be discussed.

The first use of P^3M in astrophysics was the calculation of galaxy clustering in an expanding universe by Efstathiou and Eastwood (1981). This calculation actually predates the first PM studies of this type of problem, and is also superior for the reasons noted above. For moderately clustered systems P^3M remains the method of choice, Couchman's modification is required to simulate heavily clustered systems.

1.6.4 Barnes-Hut 'Tree' method

The Barnes-Hut algorithm (Barnes and Hut, 1986), utilizes the fact that the gravitational force from a distant grouping of bodies may be approximated as the force from a point of mass equal to the total mass of the bodies acting at the center of mass of the group. This idea can be seen simply by examining the following approximation for the force, $\mathbf{F}_i$, from a group of particles of mass m_j, each at a distance r_j from particle i, with a center of mass r_{cm},

$$\mathbf{F}_i = \sum_j \frac{GM_i m_j \mathbf{r}_j}{r_j^3} = \sum_j \frac{GM_i m_j}{|\mathbf{r}_{cm} + (\mathbf{r}_j - \mathbf{r}_{cm})|^3}[\mathbf{r}_{cm} + (\mathbf{r}_j - \mathbf{r}_{cm})], \tag{1.99}$$

this step involves no approximation since the $\mathbf{r}_j$ is just expressed as the distance to the center of mass plus a short distance to the particle j. Extracting $\mathbf{r}_{cm}$ from parenthesis using a rotation matrix $\mathcal{R}$ to rotate $\mathbf{r}_{cm}$, into the direction of $\delta\mathbf{r}_j = \mathbf{r}_j - \mathbf{r}_{cm}$,

$$\mathbf{F}_i = \sum_j \frac{GM_i m_j}{r_{cm}^3 |\mathcal{I} + \frac{|\delta\mathbf{r}_j|}{|\mathbf{r}_{cm}|}\mathcal{R}|^3} (\mathcal{I} + \frac{|\delta\mathbf{r}_j|}{|\mathbf{r}_{cm}|}\mathcal{R})\mathbf{r}_{cm} \simeq \frac{GM_i(\sum_j m_j)\mathbf{r}_{cm}}{r_{cm}^3}, \tag{1.100}$$

where the approximation is true for sufficiently small $|\delta\mathbf{r}_j|/|\mathbf{r}_{cm}|$, *i.e.* the spatial extent of the mass distribution is significantly smaller than the distance to it.

To utilize this approximation it is necessary to be able to keep track of particle positions in an efficient manner. To do this an adaptive grid, known as a 'tree', is used, which is constructed in a hierarchical fashion. The tree begins with the simulation volume and then subdivides it into 2^3 units. These cells are then further subdivided in a recursive manner until the sub-cells contain one particle (in fact some cells will contain zero particles). Within this tree data-structure, local particles can be found efficiently by searching up and down the 'leafs' of the tree. Whether to treat a region as individual particles or as a collective is decided by using an 'opening angle', groups of particles appearing wider than the opening angle must be calculated on a per particle basis, while the contrary applies for smaller groups. A small opening angle leads to high accuracy but low computational efficiency.

Building the tree is an $\alpha N \log N$ operation and calculation of the forces is order $\beta N \log N$. However the operation coefficient, $\alpha + \beta$ is larger than that associated with particle-mesh (and derivative) codes. For periodic systems this fact, combined with the difficulty in calculating periodic boundaries for a tree code (images must be used), leads to a code over an order of magnitude slower than the PM and P^3M techniques.

1.6.5 Smoothed Particle Hydrodynamics

The Smoothed Particle Hydrodynamic method (SPH,Gingold and Monaghan, 1977; Lucy, 1977) allows gas forces to be calculated for particle based simulations. The method was initially developed for applications involving colliding gas polytropes but has since been applied to many different fields (see, Monaghan, 1992, for an overview). Because the method is Lagrangian it offers a number of advantages over Eulerian methods (*i.e.* concentration of the calculation in high density regions) and little effort is required to combine it with particle based gravitational solvers. This is the primary reason for its popularity in cosmology. Technical details relating to the numerical implementation of SPH are discussed in chapter 2.

At the heart of the method is the identity,

$$A(\mathbf{r}) = \int d^3r' A(\mathbf{r}')\delta(|\mathbf{r} - \mathbf{r}'|), \tag{1.101}$$

which can be approximated by,

$$< A(\mathbf{r}) >= \int d^3\mathbf{r}' A(\mathbf{r}') W(|\mathbf{r} - \mathbf{r}'|, h), \tag{1.102}$$

in the understanding that when $h \to 0$, the kernel function W becomes the delta function. In changing from the continuum limit to a discrete summation the integral is rewritten,

$$< A(\mathbf{r}) >= \int d^3\mathbf{r}' \rho(\mathbf{r}') \frac{A(\mathbf{r}')}{\rho(\mathbf{r}')} W(|\mathbf{r} - \mathbf{r}'|, h), \tag{1.103}$$

and then the discrete approximation is simply a mass-weighted sum over N neighbouring particles (provided the kernel function W has compact support),

$$\int d^3r' \rho(\mathbf{r}') \to \sum_j^N m_j.$$

Consequently, the smoothed estimate of any field can be derived simply as a weighted summation over particles. The density, for example, is given by,

$$< \rho(\mathbf{r}_i) >= \sum_{j=1}^N m_j W(|\mathbf{r}_i - \mathbf{r}_j|, h). \tag{1.104}$$

In practice, the smoothing scale h is allowed to vary for each particle so that an approximately constant number of neighbours is found (Wood, 1981). This eases the computational load in dense regions. For reasonable accuracy in calculations about 10,000 particles are necessary (see chapter 2) and approximately 50 neighbouring particles should be smoothed over to ensure stability of the integration (Steinmetz and Mueller, 1993).

1.7 Progress to date of the simulation field and comparison to observation

1.7.1 Results: Problems inherent in the standard CDM picture

Overcooling problem in CDM cosmologies

This phenomenon is also known as the 'cooling catastrophe' (White and Frenk, 1991). Since the cooling time for baryons in halos is much shorter than the free fall time, a significant proportion of the gas resides in dense cold clumps at very early epochs. This is a direct result of structure formation in a hierarchical picture, since more mass is present in low mass halos at earlier epochs. If star formation is directly related to the amount of cold gas then there will be very high star formation rates at high redshifts. Consequently, it is difficult to see how disc systems, which require baryons to dissipate convergent motions and correlate them into spin, can form. Feedback is believed to be the only method of averting the cooling catastrophe. So far only toy model simulations (such as redistributing the gas in collapsed halos once formed, Sommer-Larsen *et al.*, 1998*b*) have been shown to produce reasonable results since the star formation is turned off at the redistribution phase. As yet no self-consistent simulation (one that predicts the star formation rate and from that a distribution of SN events) has succeeded in forming a galaxy with a structure similar to those observed.

Angular momentum problem

Galaxies formed in CDM halos in SPH simulations have too little angular momentum (AM) relative to those observed. This fact was discovered at the birth of the simulation field (Navarro and Benz, 1991). It is partly a consequence of the overcooling problem, and there appear to be two mechanisms at work. Firstly, the 'Barnes Mechanism' (Barnes, 1992) progressively removes orbital angular momentum from *dense* gas cores in merging dark matter halos. The basic picture is one where during the conversion of orbital AM to internal AM (spin) of the dark matter, the gas cores are 'braked' as they over-shoot the bottom of the merging halo potentials which, via dynamical friction, results in a loss of orbital AM from the gas to the dark matter halo. If the gas can be supported higher in the halo and allowed to cool at later epochs, this problem can be avoided (Weil *et al.*, 1998). The second mechanism is due to the bar instability in disc formation. For gaseous discs, particles travelling along the bar are shocked as they meet oncoming particles from the other side of the bar. This results in the build up of large central concentrations of gas with little angular momentum (Dominguez-Tenreiro *et al.*, 1998). Significantly, this is alleviated in simulations with star formation, provided sufficient star formation occurs before or during the formation of the bar (Dominguez-Tenreiro *et al.*, 1998). Not only is the bar formation reduced, since the star particles (*i.e.* Pop II stars) diffuse away from the dense core providing stability against bar formation, the proportion of gas in the core is reduced as it is converted into stars, in turn reducing the amount of shocking. Sellwood and Moore (1999) have suggested that this bar formation is an ideal candidate for the fuel source of QSOs. Further, why QSOs 'turn off' is explicitly explained. Note that the standard artificial viscosity for SPH simulations is known to promote transfer of angular momentum, however 'shear free' variants are known (Balsara, 1995; Steinmetz, 1996) which alleviate this problem.

1.7.2 Results: Problems inherent in the numerics

The following discussion is necessarily technical in places. To avoid repetition refer to chapter 2 for an explanation of the technical details of SPH.

What to do at the high resolution end in SPH simulations?

This is a difficult issue to address since the picture is not as simple as grid based simulations, where a minimum cell size is reached. In particle based simulations the hydrodynamic resolution must match that of the gravitational solver to avoid unwanted energy transfers between the media (Bate and Burkert, 1997). This places a lower limit on the smoothing scale. However, in doing so a very significant slow down in the algorithm occurs as the SPH particles become clustered. Some authors

avoid this by allowing the gas resolution scale to become as small as it needs, leading to incorrect densities, as well as the problems previously mentioned. This is undesirable, particularly in the context of star formation where algorithms take the gas density as an input parameter, and further both the CFL and acceleration criterion on the time-step mean that as $h \to 0$, $t_{step} \to 0$, and thus simulations take tens of thousands of time-steps (Navarro and Steinmetz, 1997). An alternative solution of allowing the resolution scale used to reach a set minimum, but searching over a reduced list of neighbors, has been discussed (Navarro and White, 1993). So far this method has not been tested in detail against the method which calculates over the full neighbor list.

Does an effective Jeans' Mass need to be resolved?

It is widely known that in simulations of star formation the Jeans' Length (Mass) must always be resolved (Truelove *et al.*, 1998; Bate and Burkert, 1997). This is because star formation proceeds as the fragmentation of initially smooth clouds, for which numerical noise can be a problem—unless one ensures sufficient pressure support/resolution. This result is caused by small truncation errors in the solution growing rapidly because of the attractive nature of the gravitational force. For hierarchical simulations, there may not be a problem since growing modes are present on all length scales in the simulation. These modes have amplitudes that should swamp any error in the larger modes. Further, there may well be subtle dependence on the way the short-range forces are implemented in the gas and gravity. So far, Owen and Villumsen (1997) have conducted studies on cosmological simulations and argue that the Jeans' Mass must be resolved (although their gravitational and hydrodynamic solvers do not have the same resolution). Tittley and Couchman (1999, in prep) have found that, provided the analysed objects have formed from a sufficient number of mergers, there is negligible difference between simulations that resolve the Jeans' Mass and those which do not.

How much resolution is enough?

Most SPH work to date has not had sufficient resolution to address the formation process accurately regardless of the object under simulation. Accuracy studies (Steinmetz and Mueller, 1993) have shown that at least 10,000 particles are necessary to resolve radial shocks. Similarly, fixed grid approaches also lack resolution; very often only 30^3 or so zones of a 256^3 simulation are actually resident in the object of interest. So far only brief results from Adaptive Mesh Refinement (AMR) simulations have been published (Bryan and Kepner, 1998). The potential for these simulations is great since they actively add resolution to regions requiring it.

Just how resolution dependent are star formation algorithms?

Star formation algorithms are explicitly dependent upon the density of gas in a simulation and are thus dependent upon numerical resolution. The hope is that this dependence can be parameterized relative to the underlying resolution. Elizondo *et al.* (1999*a*) are the only group to look at resolution dependence. Their conclusions are that global properties, such as total luminosity and rotational velocity, are comparatively unaffected by resolution. However, since at higher resolution the cooling catastrophe means more gas is available for star formation, objects can become significantly bluer as star bursts become more prominent. This is roughly in line with what would be expected but ideally it should be quantifiable. So far no SPH researchers have tested resolution in detail.

1.7.3 Successes & Failures

Structure of dark matter halos

The 'Universal' dark matter profile of Navarro, Frenk & White (1996*a*) has been tested at higher and higher resolution. For a large class of halos it provides a good fit, although in simulations with 3×10^6 particles within the virial radius, Moore *et al.* (1998) find evidence for a steepening of the profile $\rho(r) \propto r^{-1.4}$. Contrary to this result, Kravstov *et al.* (1997) find evidence for a reduction of the slope in the inner core. These results are the subject of much debate since the inner core profile is derived at the resolution limit of the simulation which is susceptible to numerical artifacts.

Bias

Calculation of the bias parameter, measuring the matter density contrast relative to galaxy counts, is one of the most important challenges for hydrodynamic simulations. Accurate calculations are difficult since a large enough physical volume must be simulated and yet high resolution must be maintained to follow the galaxy formation process. Collisionless simulations (*e.g.* Jenkins *et*

al., 1998; Colin *et al.*, 1998), have given a clear demonstration of the scale dependent bias/antibias required to match observed galaxy clustering. Some Lagrangian studies (Katz *et al.*, 1996; Katz *et al.*, 1998) are limited by a small catalogue, 250 galaxies with $v_c > 100\mathrm{km\,s^{-1}}$; ($\Omega = 1$) within the simulation box, while the best Eulerian studies (Cen and Ostriker, 1998; Blanton *et al.*, 1999) are limited to a 200 kpc resolution. High resolution Lagrangian simulations are currently being analyzed (Pearce *et al.*, 1999) which increase the size of the catalogue by almost an order of magnitude. However, regardless of resolution, a clear trend is visible in these simulations: ***the correlation function of galaxies does not evolve significantly over time.*** This is in contrast to the result for the underlying dark matter where the normalization increases by a factor of 30 between $z = 3$ and $z = 0$. Thus the strong clustering of Lyman break galaxies (LBGs, see Steidel *et al.*, 1998 for a review) appears to be explained perfectly naturally within the hierarchical model. Blanton *et al.* (1999) argue that the physical origin of the static clustering amplitude is due to the following,

1. Non-linear peaks become progressively less rare over time.
2. The densest regions cannot form galaxies efficiently since the Rees-Ostriker criterion becomes significant, *i.e.* the gas in these regions becomes too hot to cool.
3. After galaxies form they are gravitationally "debiased", because their velocity field is the same as the dark matter

Disc formation is generic

The first paper to demonstrate that gaseous discs form *en masse* in simulations was that of Evrard *et al.* (1994). This result is to be expected since when radiative cooling is combined with an initial torque the out-come is always a flattened rotating system. Aside from any hydrodynamic problems, Moore *et al.* (1999) have shown sCDM produces dark matter halos with circular velocities that are too strongly centrally peaked. This has been known for a while (see the review by McGaugh 1998) and has even lead to some researchers proposing Modified Newtonian Dynamics (MOND,Milgrom, 1983*b*; Milgrom, 1983*a*) as the "correct" law of gravity.

Tully-Fisher relation

Theory on the Tully-Fisher (1977) relation, that the circular velocity v_c is related to the galaxy luminosity, L, by $v_c \propto L^{0.22}$, breaks down into two camps. One line of argument suggests that the TF is a result of self regulated star formation (*e.g.* Silk, 1997), while the other suggests that it is a direct result of the equivalence between halo mass and circular velocity (*e.g.* Mao *et al.*, 1998). Steinmetz and Navarro (1999) have shown that either the observed slope can be produced but not the zero point or vice versa (in low resolution SPH simulations with essentially no feedback from supernovae). Elizondo *et al.* (1999*b*) have published results for their multi-phase feedback simulations that show they are able to reproduce the slope *and* normalization. More observations are necessary to pin down whether the TF brightens or dims at $z < 1$. Current simulations suggest a brightening which is at odds with the majority of observations (but not all). van Campen *et al.* (1998) present a new semi-analytic+simulation method and argue that the TF relation can be reproduced provided that over-merging is corrected for.

Luminosity function of galaxies

A good fit to data is provided by the Schechter luminosity function, $dn = \phi_*(L/L_*)^\alpha \exp(-L/L_*)$, where ϕ_* and α are parameters which must be fitted. An accurate calculation of the luminosity function, *i.e.* the number of galaxies per unit volume in a given luminosity range, requires a large catalogue to draw from. Thus to date the most significant calculation is the ΛCDM simulation of Pearce *et al.* (1999). Using a Bruzual and Charlot (1993) stellar population synthesis model, they have constructed a K-Band luminosity function similar to the results of Gardener *et al.* (1997). The Pearce *et al.* luminosity function shows a clear "knee" (associated with the exponential decay) and flattening at the faint end.

Star-formation history of the Universe

The advent of deep spectroscopy has enabled the construction of a global star formation history (see the review by Steidel *et al.* (1998) and references therein). The Pearce *et al.* (1999) simulation appears to match the data well (a compilation of Lilly *et al.*, 1996; Connolly *et al.*, 1997; Madau *et al.*, 1998) although this is primarily a consequence of the large error bars on current measurements. Both better simulations, to provide a larger galaxy catalogue, and better observations, to reduce the errors, are necessary.

Does photoionization lead to reduced or increased disc masses?

There are conflicting papers on this question: Navarro and Steinmetz (1997) find that the addition of a UVX field leads to *more* compact disc galaxies (in simulations without feedback). Sommer-Larsen *et al.* (1998*b*) find the opposite (again in simulations without feedback), with the angular momentum loss relative to the dark matter being improved somewhat. This was also observed by Vedel *et al.* (1994) who used rotating sphere initial conditions. The problem may well be subtly dependent on how discs are identified within simulations. 1-d simulations by Thoul and Weinberg (1996) have shown that for halos with $v_c < 50\,\mathrm{km\,s^{-1}}$, significant suppression of baryon accumulation occurs. However, there remains a vigourous debate over the precise mass scale affected, and also the magnitude of the effect (particularly in relation to numerical resolution, Quinn *et al.*, 1996; Weinberg *et al.*, 1997).

Feedback & multi-phase models

Given the effort spent on trying to understand feedback, a solid foundation for modelling it remains elusive. The redistribution of gas following a feedback event is discussed in terms of 'blow-away' where the gas is unbound from the halo in which it resides and 'blow-out' where it is projected to the outer parts of the halo, but does not escape it. Blow-out leads to self-regularizing models where gas is recycled via a 'fountain' (White and Frenk, 1991). So far SPH simulations have failed to provide a convincing feedback model (for reasons already summarized). Some theorists would argue, in light of the complex nature of star formation, such macroscopic modelling is doomed from the start. However, as stated, the Eulerian multi-phase model used in Elizondo *et al.* (1999*a*) does appear to be able to reproduce the observed slope and normalization of the TF relation.

Imaging and photometric evolution

Using population synthesis models, such as those of Bruzual & Charlot (1993), star forming simulations can be used to provide images of simulation objects at any desired wave band (Contardo *et al.*, 1998).

1.7.4 A list of outstanding issues in galaxy formation

In summary the following phenomena/issues are currently under significant debate. Any concrete theory of galaxy formation should address all of them.

1. Shape of the galaxy luminosity function.
2. Evolution of the correlation function with redshift.
3. Tully-Fisher relation.
4. Prediction of galaxy colours.
5. Understanding of morphologies in the merger and evolution context.
6. Epoch of star formation and role of feedback, particularly global star formation history of the universe.
7. Counts of faint galaxies.
8. Origin of low surface brightness galaxies and relation to more typical systems.
9. Structure of dark matter halos.
10. Understanding of the Lyman break galaxies and whether they are progenitors of todays large spirals and ellipticals.

1.8 Layout of this thesis

Chapter 2 presents a comparison of different SPH implementations to determine whether one implementation offers significant advantages over another. A number of standard implementations are compared to newer ones, motivated by previous work. Since there are a large number of physical phenomena that contribute to galaxy formation, a variety of different simulations regimes are examined, from low density systems to high density.

Galaxy formation presents a number of theoretical and numerical challenges which are examined in chapter 3. Star formation, and how to implement it, is discussed along with a number of approaches to modelling feedback. Special attention is paid to how the energy input from feedback is distributed and, in particular, how sudden radiative losses can be prevented. The parameter space of the model is explored using a simple rotating cloud collapse model and the effect of feedback on Milky Way and dwarf galaxy prototypes is examined.

High resolution simulations require parallel computing, and chapter 4 presents work on parallelizing the 'HYDRA' simulation algorithm. Basic concepts in parallel computing are reviewed alongside a discussion of code features that increase performance on RISC processors. Coding details are discussed along with changes necessary to accommodate the data geometries used in galaxy formation. Performance is reviewed for both the standard code and the version modified for galaxy formation. Building on the work in previous chapters, chapter 5 presents results from a high resolution simulation of galaxy formation that meets a number of resolution criteria indicated in chapter 3. Conclusions and suggestions for future work are presented in chapter 2.11.

Chapter 2

A detailed examination of Smoothed Particle Hydrodynamics

"Every tool carries with it the spirit by which it had been created."

–Werner Karl Heisenberg

Results presented in this chapter were derived in collaboration with Dr E. R. Tittley, Dr F. R. Pearce, Dr H. M. P. Couchman and Dr. P. A. Thomas. This chapter appears in part as the paper "Smoothed Particle Hydrodynamics in Cosmology: A comparison of implementations", submitted to the Monthly Notices of the Royal Astronomical Society. RJT developed all the codes used within the paper. The results from simulations not conducted by RJT are summarized for completeness (sections 2.4-2.6). Of the remaining work, the cosmological test simulations (section 2.10) were run by RJT, analysed by FRP, and then written up jointly, while sections 2.7-2.9 are solely the work of RJT.

2.1 Introduction

Smoothed Particle Hydrodynamics (SPH) is a popular numerical technique for solving gas-dynamical equations. SPH is unique among numerical methods in that many algebraically equivalent – but formally different – equations of motion may be derived. In this chapter results are reported from a comparison of several implementations of SPH in tests which model physical scenarios that occur in hierarchical clustering cosmology.

SPH is fundamentally Lagrangian and fits well with gravity solvers that use tree structures (Hernquist and Katz 1989, hereafter HK89) and mesh methods supplemented by short range forces (Evrard, 1988; Couchman, Thomas and Pearce, 1995, hereafter CTP95). In an adaptive form (Wood, 1981), the algorithm lends itself readily to the wide range of densities encountered in cosmology, contrary to Eulerian methods which require the storage and evaluation of numerous sub-grids to achieve a similar dynamic range. SPH also exhibits less numerical diffusivity than comparable Eulerian techniques, and is much easier to implement in three dimensions, typically requiring 1000 or fewer lines of FORTRAN code.

The main drawback of SPH is its limited ability to follow steep density gradients and to correctly model shocks. Shock-capturing requires the introduction of an artificial viscosity (Monaghan and Gingold, 1983). A number of different alternatives may be chosen and it is not clear whether one method is to be preferred over another. The presence of shear in the flow further complicates this question.

Much emphasis has been placed upon the performance of SPH with a small number of particles (order 100 or fewer). Initial studies (Evrard, 1988) of SPH on spherical cloud collapse indicated acceptable performance when compared to low resolution Eulerian simulations, with global properties, such as total thermal energy, being reproduced well. A more recent study (Steinmetz and Muller, 1993, hereafter SM93), which compares SPH to modern Eulerian techniques (the Piecewise Parabolic Method, Collela and Woodward, 1984, and Flux-Corrected-Transport methods, Book and Boris, 1973) has shown that the performance of SPH is not as good as initially believed, and that accurate reproduction of local physical phenomena, such as the velocity field, requires as many as 10^4 particles. In the context of cosmology with hierarchical structure formation, the small-N performance remains critical as the first objects to form consist of tens – hundreds at most – of

particles and form, by definition, at the limit of resolution. It is therefore of crucial importance to ascertain the performance of different SPH implementations in the small-N regime. Awareness of this has caused a number of authors to perform detailed tests on the limits of SPH (Owen and Villumsen, 1997; Bate and Burkert, 1997). To address these concerns, some of the tests presented are specifically designed to highlight differences in performance for small N.

The goal of this chapter is to detail systematic trends in the results for different SPH implementations. Since these are most likely to be visible in the low-N regime small simulations are utilized, with larger particle numbers being used to test for convergence. Because of the importance of the adaptive smoothing length in determining the local resolution, particular attention is paid to the way in which it is calculated and updated. Efficiency of the algorithm is also a concern since real world limits on wall-clock times for simulations are impossible to avoid. Realistic hydrodynamic simulations of cosmological structure formation typically require 10^4 or more time-steps. Therefore when choosing an implementation one must carefully weigh accuracy against computational efficiency. This is a guiding principle in the investigation.

The seven tests used in this study are:

- Sod shock (section 2.4)
- Drag on a cold clump (section 2.5)
- Cooling near density jumps (section 2.6)
- Spherical collapse (section 2.7)
- Disc formation (section 2.8)
- Angular momentum transport in discs (section 2.9)
- Hierarchical structure formation (section 2.10)

Each of these tests is described in the indicated section (although sections 2.4-2.6 are only covered briefly). The tests investigate various aspects of the SPH algorithm ranging from explicit tests of the hydrodynamics to investigations specific to cosmological contexts. The Sod (1978) shock, although a relatively simple shock configuration, represents the minimum flow discontinuity that a hydrodynamic code should be able to reproduce. The spherical collapse test (Evrard, 1988), although idealised, permits an assessment of the resolution necessary to approximate spherical collapse. It also allows a comparison with other authors' results and with a high resolution spherically symmetric solution. The remaining tests are more closely tied to the arena of cosmological simulations. The cooling test looks at the problems associated with modelling different gas phases with SPH. A cold dense knot of particles embedded in a hot halo will tend to promote cooling of the hot gas because of the inability of SPH to separate the phases. The drag test looks at the behaviour of infalling satellites and the overmerging problem seen in SPH simulations (Frenk *et al.*, 1996). Finally, three tests consider the ability of the SPH algorithms to successfully model cosmic structure. Firstly, an investigation of disc formation from the collapse of a rotating cloud (Navarro and White, 1993) is conducted which is followed by an examination of the effect of angular momentum transport. These three tests are completed by an investigation of the overall distribution of hot and cold gas in a hierarchical cosmological simulation. In each case the different algorithms are compared and the reliability of the SPH method in performing that aspect of cosmological structure formation, is assessed.

The layout of the chapter is as follows. Section 2.2 reviews the basic SPH framework, including a description of a new approach developed to update the smoothing length. Next the equations of motion and internal energy are examined, along with a discussion of the procedure for symmetrization of particle forces. In sections 2.4-2.9 the test cases are presented. Each of the subsections is self contained and contains a description and motivation for the test together with results and a summary comparing the relative merits of the different implementations together with an assessment of the success with which the SPH method can perform the test. Section 2.11 briefly summarises the overall conclusions to be drawn from the test suite, indicating where each implementation has strengths and weaknesses and makes recommendations for the implementation which may be most useful in cosmological investigations.

During final preparation of this work, a preprint detailing a similar investigation was circulated by Lombardi *et al.* (1998).

2.2 Implementations of SPH

2.2.1 Features common to all implementations

All of the implementations use an adaptive particle-particle-particle-mesh (AP^3M) gravity solver (Couchman, 1991). AP^3M is more efficient than standard P^3M, as high density regions, where the particle-particle summation dominates calculation time, are evaluated using a further P^3M cycle calculated on a high resolution mesh. The process of placing a high resolution mesh is denoted 'refinement placing' and the sub-meshes are termed 'refinements'. The algorithm is highly efficient with the cycle time typically slowing by a factor of three under clustering. The most significant drawback of AP^3M is that it does not yet allow multiple time-steps, *i.e.* all the particles must be updated with the same time-step, δt. However, the calculation speed of the global solution, compared with alternative methods such as the tree-code, more than outweighs this deficiency. Full details of the adaptive scheme, in particular accuracy and timing information, may be found in Couchman (1991) and CTP95.

Time-stepping is performed using a Predict-Evaluate-Correct (PEC) scheme. This scheme is tested in detail against leapfrog and Runge-Kutta methods in CTP95. The value of the time-step, dt, is found by searching the particle lists to establish the time-step limitations of the acceleration, dt_a, and velocity arrays, dt_v. In this chapter a further time-step criterion, dt_h, is discussed, which prevents particles travelling too far within their smoothing radius, (see section 2.2.2). dt is then calculated from $dt < \kappa \min(0.4dt_v, 0.25dt_a, 0.2dt_h)$ where κ is a normalisation constant that is taken equal to unity in adiabatic simulations. In simulations with cooling, large density contrasts can develop, and a smaller value of κ is sometimes required. There is no time-step limitation for cooling since it is implemented by assuming constant density (see below and CTP95 for further details).

SPH uses a 'smoothing' kernel to interpolate local hydrodynamic quantities from a sample of neighbouring points (particles). For a continuous system an estimate of a hydrodynamic scalar $A(\mathbf{r})$ is given by

$$< A(\mathbf{r}) >= \int d^3\mathbf{r}' A(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h), \tag{2.1}$$

where h is the 'smoothing length' which sets the maximum smoothing radius and $W(\mathbf{r}, h)$, the smoothing kernel, is a function of $|\mathbf{r}|$ (note that the modulus is omitted in the kernel notation, but remains implied). For a finite number of neighbour particles the approximation to this is

$$< A(\mathbf{r}) >= \sum_j m_j \frac{A(\mathbf{r}_j)}{\rho(\mathbf{r}_j)} W(\mathbf{r} - \mathbf{r}_j, h), \tag{2.2}$$

where the radius of the smoothing kernel is set by $2h$ (for a kernel with compact support). The smoothing kernel used is the so-called B_2-spline (Monaghan and Lattanzio, 1985),

$$W(\mathbf{r}, h) = \frac{W_s(r/h)}{h^3}, \tag{2.3}$$

where if $x = r/h$,

$$W_s(x) = \frac{1}{4\pi} \begin{cases} 4 - 6x^2 + 3x^3, & 0 \le x \le 1; \\ (2-x)^3, & 1 < x \le 2; \\ 0, & x > 2. \end{cases} \tag{2.4}$$

The kernel gradient is modified to give a small repulsive force for close particles (Thomas and Couchman, 1992),

$$\frac{dW_s(x)}{dx} = -\frac{1}{4\pi} \begin{cases} 4, & 0 \le x \le 2/3; \\ 3x(4-3x), & 2/3 < x \le 1; \\ 3(2-x)^2, & 1 < x \le 2; \\ 0, & x > 2. \end{cases} \tag{2.5}$$

The primary reason for having a non-zero gradient at the origin is to avoid the artificial clustering noted by Schüssler and Schmidt (1981). (Some secondary benefits are discussed by Steinmetz, 1996.)

In standard SPH, the value of the smoothing length, h, is a constant for all particles resulting in fixed spatial resolution. Fixed h also leads to a slow-down in the calculation time when particles become clustered – successively more particles on average fall within a particle's smoothing length. In the adaptive form of SPH (Wood, 1981) the value of h is varied so that all particles have a constant (or approximately constant) number of neighbours. This leads to a resolution scale dependent upon

the local number density of particles. It also removes the slow-down in calculation time since the number of neighbours is held constant provided that the near neighbours can be found efficiently. In this chapter the desired number of neighbour particles is 52 and tests on the update algorithm used (see section 2.2.2) show that this leads to a particle having between 30 and 80 neighbours, whilst the average remains close to 50.

A minimum value of h is set by requiring that the SPH resolution not fall below that of the gravity solver. The gravitational resolution is defined to be twice the S2 softening length (Hockney and Eastwood, 1988), ϵ, as at this radius the force is closely equivalent to the $1/r^2$ law. The equivalent Plummer softening is quoted throughout the chapter, as this is the most common force softening shape. Since the minimum resolution of the SPH kernel is the diameter of the smallest smoothing sphere, $4h_{min}$, equating this to the minimum gravitational resolution yields

$$2\epsilon = 4h_{min}. \tag{2.6}$$

Unlike other authors, once this minimum h is reached by a particle, smoothing occurs over *all* neighbouring particles within a radius of $2h_{min}$. As a result, setting a lower h_{min} increases efficiency as fewer particles contribute to the sampling, but this leads to a mismatch between hydrodynamic and gravitational resolution scales which is undesirable and may cause spurious effects (Bate and Burkert, 1997; Sommer-Larsen *et al.*, 1998*a*).

When required, radiative cooling is implemented in an integral form that assumes constant density over a time-step (Thomas and Couchman, 1992). The change in the specific energy, e, is evaluated from

$$\int_e^{e-\Delta e} \frac{de}{\Lambda} = -\frac{n_i^2}{\rho_i}\Delta t, \tag{2.7}$$

where Δt is the time-step, n_i the number density and $n_i^2\Lambda$ is the power radiated per unit volume. In doing this the time-step limitation for cooling is circumvented.

2.2.2 An improved first-order smoothing-length update algorithm

As stated earlier, in the adaptive implementation of SPH the smoothing length, h, is updated each time-step so that the number of neighbours is held close to, or exactly at, a constant. Ensuring an exactly constant number of neighbours is computationally expensive (requiring additional neighbour-list searching) and hence many researchers prefer to update h using an algorithm that is closely linked to the local density of a particle.

Two guiding principles have been adopted in the design of a new algorithm. First, since gravitational forces are attractive, the algorithm will spend most of its time decreasing h (void evolution is linear and hence smoothing lengths update slowly in these regions). Second, smoothing over more particles than the desired number, N_{smooth}, is generally preferable to smoothing over fewer. Despite some loss of spatial resolution and computational efficiency, it does not 'break' the SPH algorithm. Smoothing over too few particles can lead to unphysical shocking.

A popular method for updating the smoothing length is to predict h at the next time-step from an average of the current h and the h implied by the number of neighbours found at the current time-step (HK89, note there is a further step to the full HK89 h-update algorithm – see later). This is expressed as

$$h_i^n = \frac{h_i^{n-1}}{2}\left[1 + \left(\frac{N_s}{N_i^{n-1}}\right)^{1/3}\right], \tag{2.8}$$

where h_i^n is the smoothing length at time-step n for particle i, N_s is the desired number of neighbours and N_i^{n-1} is the number of neighbours found at step $n-1$. The performance of this algorithm for a rotating cloud collapse problem (see section 2.8) is shown in the top left panel of figure 2.1. The rotating cloud collapse is a difficult problem for the h update algorithm to follow since the geometry of the cloud rapidly changes from three to two dimensions. The plot shows that both the maximum and minimum number of neighbours (selected from all the SPH particles in the simulation) exhibit a significant amount of scatter. Clearly, the maximum number of neighbours increases rapidly once the $h = h_{min}$ limit is reached.

The rotating cloud problem demonstrates how this algorithm may become unstable when a particle approaches a high density region (the algorithm is quite stable where the density gradients are small). If the current h is too large, the neighbour count will be too high leading to an underestimate of h. At the next step too few neighbours may be found. The result is an oscillation in the estimates of h and in the number of neighbours as a particle accretes on to a high density region from a low density one. This oscillatory behaviour is visible in the fluctuations of the minimum number of neighbours in top left panel of figure 2.1.

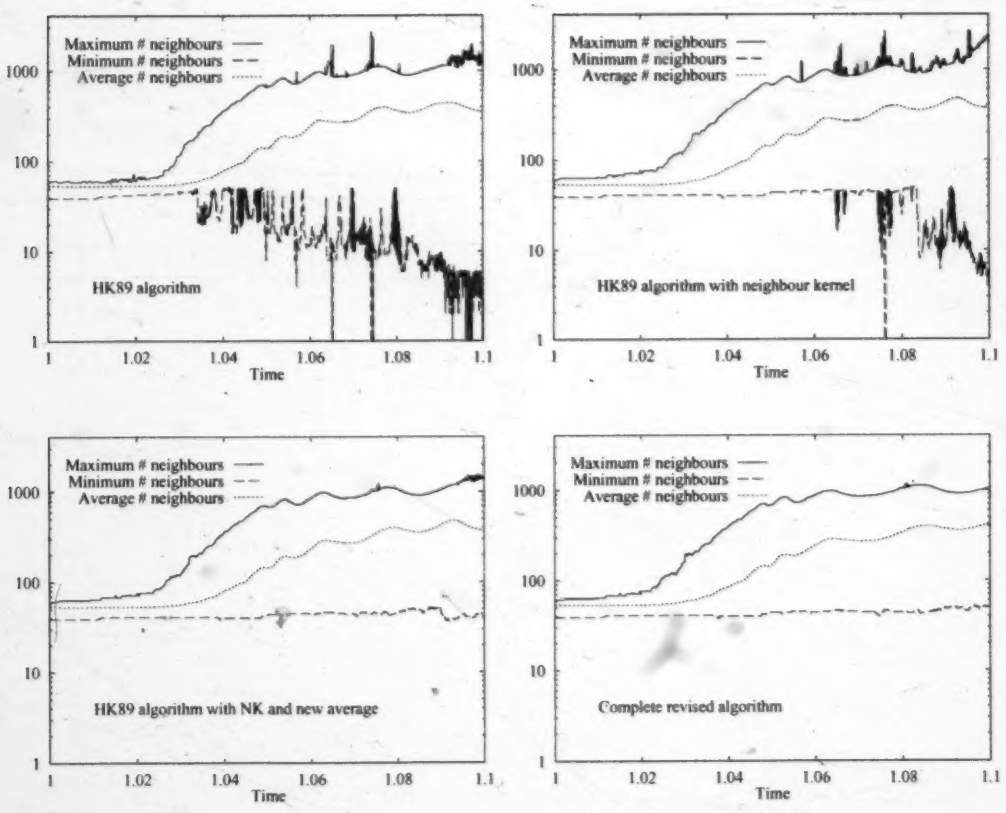

Figure 2.1: Improvement in the neighbour counts as each component of the new algorithm is added. A rotating cloud collapse problem (see section 2.8) was repeated with each of our modifications to the HK89 algorithm. Time units are that of the approximate free-fall time $(R^3/GM)^{1/2}$. The improvement from the first panel (upper left) to the third panel (bottom left) is clear, the final panel shows a slight reduction in the range of neighbours and in the oscillation of the counts. The large increase in the maximum number of neighbours is due to the h_{min} limit being reached.

The instability can be partially cured by using an extremely small time-step. However, this is not practical as simulations currently take thousands of time-steps. A second alternative is to iteratively find the correct h and recalculate the necessary hydrodynamic quantities – this is the second step in the full h-update algorithm presented in HK89. Whilst an effective solution, it is clearly inefficient. An algorithm that predicts the desired h correctly is always preferable.

A solution suggested by Wadsley (1997) that helps alleviate the discontinuity in the number of neighbours, is to 'count' neighbours with a smoothed kernel rather than the usual tophat. This (normalised) weighted neighbour count is then used in the h update equation rather than N_i^{n-1}. The instability in the standard HK89 algorithm is caused by the sharp discontinuity in the tophat at $r = 2h$. Hence it is required that the new kernel smoothly approach 0 at $2h$. Secondly, the most local particles within the smoothing radius should be counted at full weight. These considerations suggest a kernel that is unity to a certain radius followed by a smooth monotonic decrease to zero. The following function is chosen,

$$W_{nn}(r/h) = \begin{cases} 1, & 0 \le r/h < 3/2; \\ \pi W_s(4(r/h - 3/2)), & 3/2 \le r/h \le 2, \end{cases} \tag{2.9}$$

where $W_s(x)$ is the normalised B_2-spline kernel. Experiments were conducted to determine the optimal radius for switching over to the spline and a value of $r = 3h/2$ has proven to be optimal. This value provides a good balance between the smoothness of the variation of the estimate and the closeness of actual number of neighbours to the desired number. At this value approximately half of the smoothing volume is counted at full weight. For smaller values the smoothed estimate becomes progressively more unreliable. Conversely as the limiting radius is increased the gradient of the kernel at $r \simeq 2h$ becomes too steep. The improvement in fluctuation of the maximum and minimum number of neighbours can be seen in the top right panel of figure 2.1.

The next step in the construction of the new algorithm is to adjust the the average used to update h. The primary reason for doing this is to avoid sudden changes in the smoothing length. Whilst the neighbour counting kernel helps to alleviate this problem, it does not remove it entirely. Secondly, since the time-step will be limited by only allowing the particles to move a certain fraction of h, it is also useful to limit the change in h. The motivation here is that a particle which is approaching a high density region, for example, and is restricted to move $0.2h$ per time-step, should only be allowed to have $h \to 0.9h$ (since the smoothing radius is $2h$). However, a particle at the center of homologous flow must be able to update faster since collapse occurs from all directions. To account for this it is helpful to permit a slightly larger change in h, $h \to 0.8h$, for example.

Setting $s = (N_s/N_i^{n-1})^{1/3}$, equation 2.8 may be expressed as

$$h_i^n = h_i^{n-1}(1 - a + as), \tag{2.10}$$

where a is a weighting coefficient, and for equation 2.8, $a = 0.5$. Test were conducted on the performance of this average for $a \in [0.2, 0.5]$. A value of $a = 0.4$ proved optimal, reducing scatter significantly yet allowing a sufficiently large change in h. A problem remains that if $s \simeq 0$ then $h_i^n = 0.6h_i^{n-1}$, which represents a large change if the time-step is limited according to an h/v criterion. Hence the scheme was implemented with the asymptotic property $h_i^n = 0.8h_i^{n-1}$, but for small changes in h it yields $h_i^n = h_i^{n-1}(0.6 + 0.4s)$. The function used for determining the weighting variable a is

$$a = \begin{cases} 0.2(1 + s^2), & s < 1; \\ 0.2(1 + 1/s^3), & s \ge 1. \end{cases} \tag{2.11}$$

A plot of this function compared to the 0.6,0.4 weighted average can be seen in figure 2.2. The lower left panel of figure 2.1 shows that introduction of this average reduces the scatter in both the maximum and minimum number of neighbours.

Even with the improvements made so far, it remains possible that the particle may move very quickly on to a high density region causing a sudden change that cannot be captured by the neighbour counting kernel. Thus, it is sensible to limit the time-step according to a criterion which is the smoothing radius divided by the highest velocity of the neighbour particles searched over. Further, the velocity $\mathbf{v}_r$, must be measured within the frame of the particle under consideration, $\mathbf{v}_r = \mathbf{v}_i - \mathbf{v}_j$. The new time-step criterion may be summarised,

$$dt = C_h \min_i(h_i / \max_{j_{neb}}(|\mathbf{v}_i - \mathbf{v}_{j_{neb}}|), \tag{2.12}$$

where C_h is the Courant number and the i, j_{neb} subscripts denote a reduction over all the particles i, and the neighbour particles j_{neb}. $C_h = 0.2$ was chosen, which is usually the limiting factor in time-step selection. The gains from introducing this condition are marginal for the neighbour count in this test (maximum and minimum values are within a tighter range and show slightly less oscillation).

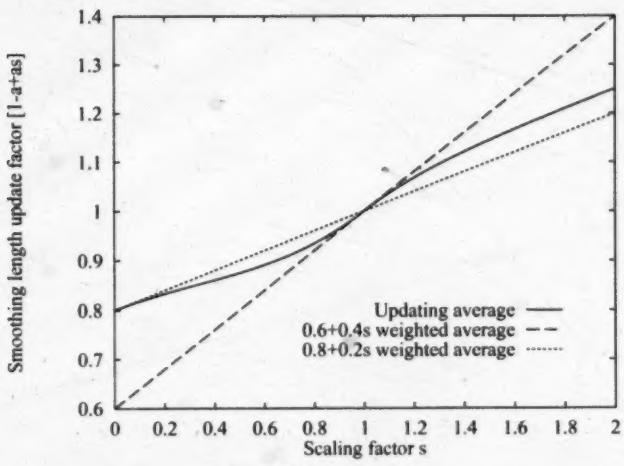

Figure 2.2: Weighting function compared to simple weighted averages.

When taken together all of these adjustments combine to make a scheme that is both fast and very stable. The final result of including these improvements is seen in the lower right hand panel of figure 2.1.

2.2.3 Equations of motion

The SPH equation of motion is derived from

$$\frac{dv}{dt} = -\frac{1}{\rho}\nabla P, \tag{2.13}$$

using identities involving the pressure and density. An excellent review of this derivation, and why so many different schemes are possible, may be found in Monaghan (1988). In short, different identities produce different equations of motion and a clear demonstration of this can be seen by comparing the equations of motion of HK89 to those of SM93.

Once an adaptive scheme is implemented in SPH, neighbour smoothing develops a dualism – the smoothing may be interpreted as either a "gather" or a "scatter" process. If smoothing is conducted at the position of particle i, then the contribution of particle j to the smoothed estimate may be evaluated using the value of h from either particle i (gather) or particle j (scatter). For constant h SPH the schemes are equivalent.

The question of which neighbour smoothing method to use may be circumvented by using a formalism based upon the average of the two smoothing lengths (Evrard, 1988; Benz, 1990). Under this prescription, the gather and scatter interpretations are equivalent since the smoothing length h_{ij} is the same whether evaluated for particle i or for particle j. This prescription is denoted h-averaging. The density estimate under this prescription is given by,

$$< \rho(\mathbf{r}_i) >= \sum_{j=1, r_{ij}<2h_{ij}}^{N} m_j W(\mathbf{r}_i - \mathbf{r}_j, h_{ij}), \tag{2.14}$$

where,

$$h_{ij} = (h_i + h_j)/2, \tag{2.15}$$

and the $r_{ij} < 2h_{ij}$ qualifier on the neighbour summation denotes the search is conducted over particles for which $r_{ij} < 2h_{ij}$.

Most h-averaging schemes use the arithmetic mean of the smoothing lengths but it is possible to consider other averages (any 'average' that is symmetric in $i - j$ is potentially acceptable). Two other averages of interest are the geometric mean and the harmonic mean. Note that the harmonic and geometric means are zero for $s = 0$ whereas the arithmetic mean is one-half. This has potentially important consequences when particles with large h interact with particles that have a small h. This

situation occurs at the boundary of high density regions in simulations with radiative cooling (see section 2.6 for a brief discussion of this).

An alternative way of circumventing the question of whether to use the gather or scatter interpretation is to combine them into one hybrid framework by averaging the kernels. In this scheme the density estimate is given by,

$$< \rho(\mathbf{r}_i) >= \sum_{j=1, r_{ij}<2h_{ij}}^{N} m_j[W(\mathbf{r}_i - \mathbf{r}_j, h_i) + W(\mathbf{r}_i - \mathbf{r}_j, h_j)]/2, \tag{2.16}$$

and this scheme is denoted kernel averaging. The averaged kernel then replaces the normal kernel in all equations. Normally the neighbour search is conducted over particles for which $r_{ij} < 2\max(h_i, h_j)$, but this was not adopted since it was desired to keep the differences between the algorithms as small as possible (within the frame work of changing the symmetrization procedure). In practice the effect of changing the neighbour search should be small since the differences occur at large radii where the contribution of the kernel is small. Rewriting equation 2.13 using the identity

$$\frac{\nabla P}{\rho} = \nabla \frac{P}{\rho} + \frac{P}{\rho^2}\nabla\rho, \tag{2.17}$$

the SPH equation of motion with kernel averaging becomes,

$$\frac{d\mathbf{v}_i}{dt} = -\sum_{j=1, r_{ij}<2h_{ij}}^{N} m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2}\right) \times$$

$$\nabla_i[W(\mathbf{r}_i - \mathbf{r}_j, h_i) + W(\mathbf{r}_i - \mathbf{r}_j, h_j)]/2. \tag{2.18}$$

This equation of motion is used in SM93 (with the different neighbour search mentioned).

Thomas and Couchman (1992, hereafter TC92), present an alternative prescription where the force term is symmetrized and the density is calculated under the gather interpretation. Using the standard pressure and density identity in equation 2.17, the acceleration for particle i is written,

$$\frac{d\mathbf{v}_i}{dt} = \sum_{j=1}^{N} \frac{\mathbf{f}_{ij}}{m_i} = -\sum_{j=1, r_{ij}<2h_i}^{N} m_j \frac{P_i}{\rho_i^2}\nabla_i W(\mathbf{r}_i - \mathbf{r}_j, h_i)$$

$$+ \sum_{j=1, r_{ij}<2h_j}^{N} m_j \frac{P_j}{\rho_j^2}\nabla_j W(\mathbf{r}_i - \mathbf{r}_j, h_j). \tag{2.19}$$

This symmetrization is used in the current implementation of the publicly available code, HYDRA. When combined with an artificial viscosity that does not require the pre-computation of all density values this scheme is extremely efficient.

In deriving equation 2.19 the approximation

$$\nabla_i W(\mathbf{r}_i - \mathbf{r}_j, h_i) \simeq -\nabla_j W(\mathbf{r}_i - \mathbf{r}_j, h_j) \tag{2.20}$$

was used. This approach to symmetrization of the equation of motion is fundamentally different to the other two schemes which both result in a kernel symmetric in i and j, and involve no approximation. The substitution is not correct to first order in h and may introduce small errors. If this symmetrization is supplemented by either h-averaging or kernel averaging (there is no argument against this) then the substitution is correct to first order in h (but not in ∇h).

2.2.4 Internal energy equation

The SPH internal energy equation is derived from

$$\frac{d\epsilon}{dt} = -\frac{P}{\rho}\nabla.\mathbf{v}. \tag{2.21}$$

The SPH estimate for $\nabla.\mathbf{v}$ may be used to calculate $d\epsilon/dt$ directly, yielding the internal energy equation for particle i,

$$\frac{d\epsilon_i}{dt} = -\frac{P_i}{\rho_i}\nabla.\mathbf{v}_i. \tag{2.22}$$

Explicitly writing the summation for the SPH estimate gives

$$\frac{d\epsilon_i}{dt} = \sum_j m_j \frac{P_i}{\rho_i^2} (\mathbf{v}_i - \mathbf{v}_j).\nabla_i W(\mathbf{r}_i - \mathbf{r}_j, h). \tag{2.23}$$

Adding kernel averaging and inserting the artificial viscosity $P_i \rightarrow P_i + \rho_i^2 \Pi_{ij}/2$ (see equation 2.26), yields the internal energy equation used in SM93. Whilst not strictly compatible with equation 2.18 (equation 2.23 has no dependence upon P_j) it can nevertheless be shown that energy will be conserved (Benz, 1990).

An internal energy equation may be constructed using the same symmetrization as equation 2.19, yielding

$$\frac{d\epsilon_i}{dt} = \frac{1}{2} \sum_{j=1, r_{ij}<2h_i}^{N} m_j (\mathbf{v}_i - \mathbf{v}_j).\frac{P_i}{\rho_i^2} \nabla_i W(\mathbf{r}_i - \mathbf{r}_j, h_i)$$

$$-\frac{1}{2} \sum_{j=1, r_{ij}<2h_j}^{N} m_j (\mathbf{v}_i - \mathbf{v}_j).\frac{P_j}{\rho_j^2} \nabla_j W(\mathbf{r}_i - \mathbf{r}_j, h_j). \tag{2.24}$$

A similar equation was considered in TC92, and was shown to exhibit excellent energy conservation. In CTP95 the entropy scatter produced by this equation was compared to that of equation 2.22 and was shown to be significantly larger. Therefore to avoid this problem only the energy equation 2.23 is considered.

It has been shown by Nelson and Papaloizou (1993,1994) and Serna, Alimi and Chieze (1996), that inclusion of the ∇h terms, in both the equation of motion and internal energy, is necessary to ensure accurate conservation of both energy and entropy (to within 0.5%). Due to the overhead involved in computing these terms they are neglected in this investigation. Evrard (1988) presents an analysis of why these terms are expected to be small.

2.2.5 Artificial viscosity algorithms

An artificial viscosity is necessary in SPH to dissipate convergent motion and hence prevent interpenetration of gas clouds (Monaghan and Gingold, 1983).

Four different types of artificial viscosity are investigated. The first considered is the implementation of TC92, where an additional component is added to a particle's pressure,

$$P_i \rightarrow \begin{cases} P_i + \rho_i[-\alpha c_i h_i \nabla.\mathbf{v}_i + \beta(h_i \nabla.\mathbf{v}_i)^2], & \nabla.\mathbf{v}_i < 0; \\ P_i, & \nabla.\mathbf{v}_i \geq 0, \end{cases} \tag{2.25}$$

where c_i is the sound speed for the particle. Typically, the viscosity coefficients are $\alpha = 1$ and $\beta = 2$. This algorithm is a combination of the Von Neumann-Richtmeyer and bulk viscosities (see Gingold and Monaghan, 1983). A notable feature is that it uses a $\nabla.\mathbf{v}_i$ 'trigger' so that it only applies to particles for which the local velocity field is convergent. This is different from most other implementations, which use an $\mathbf{r}_{ij}.\mathbf{v}_{ij} < 0$ trigger. With this artificial viscosity the first term in equation 2.19 does not depend upon the density of particle j. This is advantageous numerically since it is not necessary to construct the neighbour lists twice. In this circumstance one can reduce the SPH search over particles to a primary loop, during which the density is calculated and the neighbour list is formed and stored, followed by a secondary loop through the stored neighbour list to calculate the forces and internal energy.

A modification of this artificial viscosity is to estimate the velocity divergence over a smaller number of neighbours found by searching to a reduced radius ($h/\sqrt{2}$). This was motivated by the observation that in some circumstances the gas does not shock as effectively as when a pairwise viscosity is employed. The reduced search radius leads to a higher resolution (but likely noisier) estimate of $\nabla.\mathbf{v}$.

The third artificial viscosity considered is the standard 'Monaghan' artificial viscosity (Monaghan and Gingold, 1983). This artificial viscosity has an explicit $i - j$ particle label symmetry which is motivated to fit with equation 2.18. In this algorithm the summation of P/ρ^2 terms in equation 2.18 is extended to include a term Π_{ij}, which is given by

$$\Pi_{ij} = \frac{-\alpha \mu_{ij} \bar{c}_{ij} + \beta \mu_{ij}^2}{\bar{\rho}_{ij}}, \tag{2.26}$$

where

$$\mu_{ij} = \begin{cases} \bar{h}_{ij}\mathbf{v}_{ij}.\mathbf{r}_{ij}/(r_{ij}^2 + \nu^2), & \mathbf{v}_{ij}.\mathbf{r}_{ij} < 0; \\ 0, & \mathbf{v}_{ij}.\mathbf{r}_{ij} \geq 0, \end{cases} \tag{2.27}$$

with the bar denoting arithmetic averaging of the quantities for particles i and j and $\nu^2 = 0.01\bar{h}_{ij}^2$ is a term included to prevent numerical divergences. Again, typical values for the coefficients are $\alpha = 1$ and $\beta = 2$ although for problems involving weak shocks values of 0.5 and 1, respectively, may be preferable. This artificial viscosity has the same functional dependence upon h as the $\nabla.\mathbf{v}$ version. Since this algorithm utilises a pairwise evaluation of the relative convergence of particles, it is capable of preventing interpenetration very effectively. A drawback is that it requires the neighbour list for particles to be calculated twice since the force on particle i depends explicitly on the density of particle j (storing all the neighbour lists is possible, but in practice too memory consuming).

One major concern about this algorithm relates to its damping effect on angular momentum. In the presence of shear flows, $\nabla.\mathbf{v} = 0$, $\nabla \times \mathbf{v} > 0$ the pairwise $\mathbf{r}_{ij}.\mathbf{v}_{ij}$ term can be non zero and hence the viscosity does not vanish. This leads to a large shear viscosity which is highly undesirable in simulations of disc formation, for example. A way around this problem is to add a shear-correcting term to the artificial viscosity (Balsara, 1995). The modification used here is given by Steinmetz (1996);

$$\Pi_{ij} \rightarrow \tilde{\Pi}_{ij} = \Pi_{ij}\bar{f}_{ij}, \tag{2.28}$$

where,

$$\bar{f}_{ij} = \frac{f_i + f_j}{2}, \tag{2.29}$$

and,

$$f_i = \frac{|<\nabla.\mathbf{v}>_i|}{|<\nabla.\mathbf{v}>_i| + |<\nabla \times \mathbf{v}>_i| + 0.0001c_i/h_i}. \tag{2.30}$$

For pure compressional flows $f = 1$ and the contribution of Π_{ij} is unaffected whilst in shear flows $f = 0$ and the viscosity is zero. This modification has been studied by Navarro and Steinmetz (1997) who found that the dissipation of angular momentum is drastically reduced in small-N problems using this method. However, of concern is whether this modification leads to poorer shock-capturing. This may arise due to sampling error in the SPH estimates of the velocity divergence and curl. To test this hypothesis the shear-corrected viscosity is compared to the standard Monaghan viscosity.

The efficiency gained from having an artificial viscosity that does not depend upon the density of particle j motivated the following modification of the Monaghan viscosity: use the same quantities as equation 2.26, except that,

$$\bar{\rho}_{ij} \rightarrow \tilde{\rho}_{ij} = \rho_i(1 + (h_i/h_j)^3)/2, \tag{2.31}$$

which provides an estimate of $\bar{\rho}_{ij}$. The estimate is based upon the approximation $\rho_j \simeq \rho_i h_i^3/h_j^3$. A plot of this estimate against the real ρ_j in the spherical collapse test (see section 2.7) shows the maximum error to be a factor of ten. In practice the maximum error is dependent upon the problem being studied, but in general the error usually falls in the range of a factor of two. Note that the shear-free single-sided Monaghan variant would have to implemented as,

$$\tilde{\Pi}_{ij} = \frac{-\alpha\mu_{ij}\bar{c}_{ij} + \beta\mu_{ij}^2}{\tilde{\rho}_{ij}} f_i, \tag{2.32}$$

thereby removing the dependence upon ρ_j and f_j. Despite the estimated quantity $\tilde{\rho}_{ij}$, this artificial viscosity, when used in conjunction with the equation of motion in equation 2.19, still results in a momentum-conserving scheme.

In principle the $\nabla.\mathbf{v}$-based artificial viscosity should not suffer the problem of damping during pure shear flows, since the artificial viscosity only acts in compressive flows. A useful test is to supplement this artificial viscosity with the shear-correction term. This enables an estimate to be made of the extent to which the correction term under-damps due to SPH sampling errors.

2.3 Summary of implementations

The SPH implementations that were examined are listed in Table 2.1. The list, whilst not exhaustive, represents a range of common variants for the SPH algorithm. The motivation for the investigation is to determine the extent to which the different implementations affect behaviour in cosmological settings. The full h-updating scheme described in section 2.2.2 was used in all cases.

Version	Artificial Viscosity	symmetrization	equation of motion
1	TC92	TC92	TC92
2	TC92+shear correction	TC92	TC92
3	local TC92	TC92	TC92
4	Monaghan	arithmetic h_{ij}	SM93
5	Monaghan	harmonic h_{ij}	SM93
6	Monaghan	kernel averaging	SM93
7	Monaghan+shear correction	arithmetic h_{ij}	SM93
8	Monaghan+shear correction	harmonic h_{ij}	SM93
9	Monaghan+shear correction	kernel averaging	SM93
10	Monaghan	TC92	TC92
11	Monaghan $\tilde{\rho}_{ij}$	TC92	TC92
12	Monaghan $\tilde{\rho}_{ij}$	TC92 + kernel av	TC92

Table 2.1: Summary of the implementations examined. 'Monaghan' is the artificial viscosity of equation 2.26. Steinmetz-type shear-correction (Steinmetz, 1996) is applied where noted. The remaining terms are discussed in the text.

2.4 Shock tube: Summary

The shock-tube test was conducted by modelling a contact discontinuity between a region of high pressure gas and a region of low pressure gas. This test is known as the 'Sod' shock (Sod, 1978) and analytic solutions for the evolution are presented in Hawley, Smarr and Wilson (1984) and Rasio and Shapiro (1991). It was found that the codes divided into two groups, with one group (4-12) being more accurate than the other. Versions 1-3 produced broader shocks because the averaged viscosity used in them is unable to damp coincident motion effectively. In contrast the pairwise artificial viscosity in versions 4-12 is extremely effective at maintaining an ordered flow and thus enables the shock to be resolved more accurately. Shock widths were approximately $2h$ for versions 4-12, and $3h$ for versions 1-3. The addition of shear-correction to the artificial viscosity was found to degrade the shock capturing ability, but the effect was much less significant than changing the viscosity to the averaged version. This work was conducted by Dr F. R. Pearce.

2.5 Drag Test: Summary

The drag test examined the slowing of a cold knot of gas due to the presence of a warm ambient medium. The supersonic, transonic and subsonic motion realms were all examined to see if different behaviours were exhibited. The test is of interest to cosmology since drag may exacerbate the overmerging problem in gas dynamic simulations (see Frenk *et al.*, 1996). For all implementations it was found that in the supersonic and transonic regimes the drag values were close (within 10% in the higher resolution studies) to an analytic estimate based on a particle sweeping up mass as it travelled through the medium and consequently, via conservation of momentum, losing velocity. The subsonic tests showed a much higher breaking effect, suffering at least 50% more deceleration than predicted. There was little difference between the implementations in the supersonic and transonic regime, although results tended to produce lower drag for the h-averaging prescriptions. In the subsonic tests the Monaghan viscosity coupled to the TC92 symmetrization performed badly (versions 10 & 11), although the addition of kernel averaging removed this problem (version 12). Adding shear correction was found to reduce drag in the subsonic test by 20-30%, but had little effect elsewhere. This work was conducted by Dr E. R. Tittley.

2.6 Cooling test: Summary

The cooling test was motivated by the observation that in high resolution simulations gas particles at the edge of high density regions undergo rapid (artificial) cooling. The cause of this problem is that these particles smooth over the high density region, in turn increasing their density values and subsequently increasing the cooling rate. The test was set up by placing a very dense knot of gas in an ambient medium and then examining the evolution of the system. Interpretation was somewhat

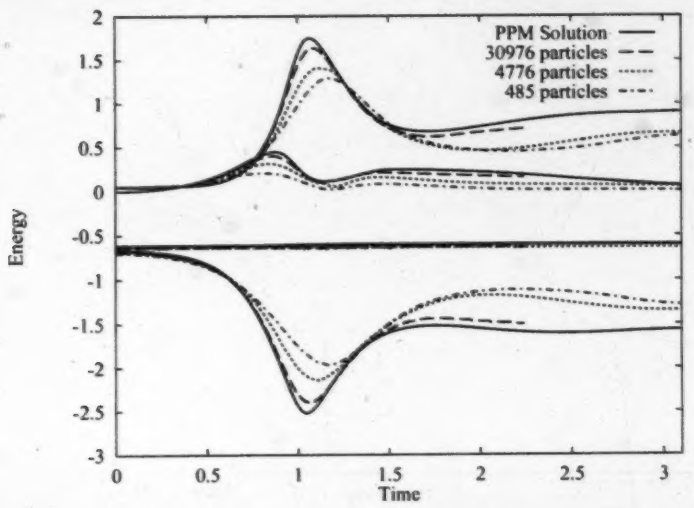

Figure 2.3: Convergence of energy values with particle number for the Evrard collapse test using version 12. Energy is plotted on the y axis and time along the x axis in normalised units. The sets of curves are, from top to bottom, the thermal, kinetic, total and potential energies. There is comparatively little difference between the 4776 and 485 particle collapse because they were run with the same softening length. The 30796 particle run very accurately matches the solution calculated in SM93.

difficult since the system had to first relax before reaching a quasi-steady-state where the effect could be measured accurately. All of the implementations exhibited the overcooling phenomenon and one implementation, version 11, performed worse than the rest having approximately 60% more cooled mass than the average. The remaining implementations lay in a band $\pm 10\%$ about the average. This work was conducted by Dr E. R. Tittley.

2.7 Spherical collapse

This test examines the adiabatic collapse of an initially isothermal spherical gas cloud. This test is denoted the 'Evrard collapse' since it first appears in Evrard (1988), and since then has proved itself as a useful test case for combined SPH-gravity codes (e.g. HK89; SM93; Serna *et al.*, 1996; Hultman and Kallander 1997).

2.7.1 Units and initial conditions

Test results are presented in normalised units. The density, internal energy, velocity, pressure and time are normalised by $3M/4\pi R^3$, GM/R, $(GM/R)^{\frac{1}{2}}$, ρu and $(R^3/GM)^{\frac{1}{2}}$, respectively, where R denotes the initial radius and M the total mass. The initial physical density distribution is given by

$$\rho(r) = \frac{M(R)}{2\pi R^2}\frac{1}{r}, \tag{2.33}$$

which is achieved by applying a radial stretch to an initially uniform grid (Evrard, 1988). This configuration is to be preferred over a random one since it has significantly less sampling error than a random distribution (Hultman and Kaellander, 1997). The internal energy of the system was chosen to be $0.05GM/R$, and the adiabatic index $\gamma = 5/3$. The softening length used for each simulation is given in the caption to Table 2.2.

2.7.2 System evolution

The evolution of the energy of system is shown in figure 2.3. As the collapse occurs the gas is heated until the temperature of the core rises sufficiently to cause a 'bounce', after which a shock propagates outward through the gas. After the shock has passed through the majority of the gas, the final state of the system is one of virial equilibrium. Along with placing emphasis on the performance of the implementations for small-N systems, the convergence at larger values of N is examined. A summary of the runs performed is presented in Table 2.2.

2.7.3 Results from the 485 particle collapse

With the N considered here, the standard HYDRA code, and variants of it, do relatively poorly in this test. A lack of thermalization is clearly visible in figure 2.6, where the differences in energy between versions 1–3 of the code and version 12 are shown along the top row. These implementations all have thermal energy peaks that are half that of the other codes. The other versions perform reasonably similarly with minor differences being seen in the peak thermal energy and in the strength of post bounce oscillation (note the change in y-axis scale on the bottom two rows of figure 2.6).

The modified viscosity variant, 3, is slightly better at thermalization than versions 1 and 2 but still much worse than any of the other versions. This indicates that the artificial viscosity is the primary factor in deciding the amount of kinetic energy that is thermalised (as expected). The more local estimate of $\nabla.\mathbf{v}$ used in version 3 captures the strong flow convergence near the bounce better than the standard estimate, leading to greater dissipation. At the other extreme the pairwise Monaghan viscosity, which uses the $\mathbf{r}.\mathbf{v}$ trigger, leads to far more dissipation at bounce. It is important to note that the final energy values for the virialised state are very similar for all codes, even though their evolution is very different in some cases.

Similar characteristics can be seen in the kinetic energy graphs. The gas in versions 1–3 develops very little kinetic energy. The primary cause of this is the $\nabla.\mathbf{v}$ artificial viscosity which trips during the early stages of collapse (prior to t=0.6) and causes an increase in the thermal energy for all particles. This acts to decrease the compressibility of the gas. For all the other codes which use the Monaghan viscosity (or variant), the $\mathbf{r}.\mathbf{v}$ trigger produces far less dissipation during the early stages of evolution, and the gas develops more kinetic energy.

For the $N = 485$ test, figures 2.6, 2.5 and 2.4 demonstrate that there is no clear optimal implementation, but the general comparison of versions 1, 3, 7 and 12 in figure 2.5, indicates that some perform marginally better than others. Notable features of the high resolution radial PPM solution in SM93 are a strong initial peak in the thermal energy and little post bounce oscillation. If a model is chosen on the basis of these criteria then version 12 performs best, although it is difficult to differentiate versions 4–12 in figure 2.6. Version 12 has both a high initial peak and very little post bounce oscillation. It also conserves energy well. The shear-correction term does have some effect (middle row of figure 2.6), in agreement with the observations in section 2.4. The general influence of the shear-correction is to increase the peak thermal energy at bounce and introduce slightly more post-bounce oscillation, although, again, this is not a significant effect. The term has little effect on the radial profile.

The effect of $\tilde{\rho}_{ij}$ replacement can be seen in the bottom row of figure 2.6. Versions 10 and 11 differ only by this substitution and there is very little to choose between them, the scatter between the kernel-averaged version 6 and version 12 being as large. None of the implementations considered show poor energy conservation, and all show excellent conservation of angular momentum.

2.7.4 Effect of numerical resolution

Increasing N to 4776 produces the expected results. The implementations with pairwise artificial viscosity converge to very similar energy profiles, see figures 2.7 and 2.8. The 10% spread seen in the $N = 485$ test is reduced to close to 1% and the limited scatter visible in the radial profiles is further reduced. The shear-correction term also has much less effect on the radial energy profile. For comparison, figure 2.3 shows the convergence of runs performed with different particle number using code version 12. This plot should be compared with figure 6 of SM93. Clearly for a Monaghan-type viscosity the differences caused by particle number and softening parameter are much larger than those caused by the choice of SPH implementation for this range of particle number.

Versions 1, 3, 4, 6, 11 and 12 were run with 30976 particles to check for convergence of the implementations at high resolution. Radial profiles at t=1.4 are plotted in figure 2.4. It is evident from these profiles that the solutions are much closer than the radial profiles for the 485 particle collapse. However, the difference between the versions with the standard hydra viscosity and the Monaghan variants remains comparatively large – a factor of two at the center in the pressure and density at t=1.4. (The relatively large energy error is a product of our choosing a longer timestep

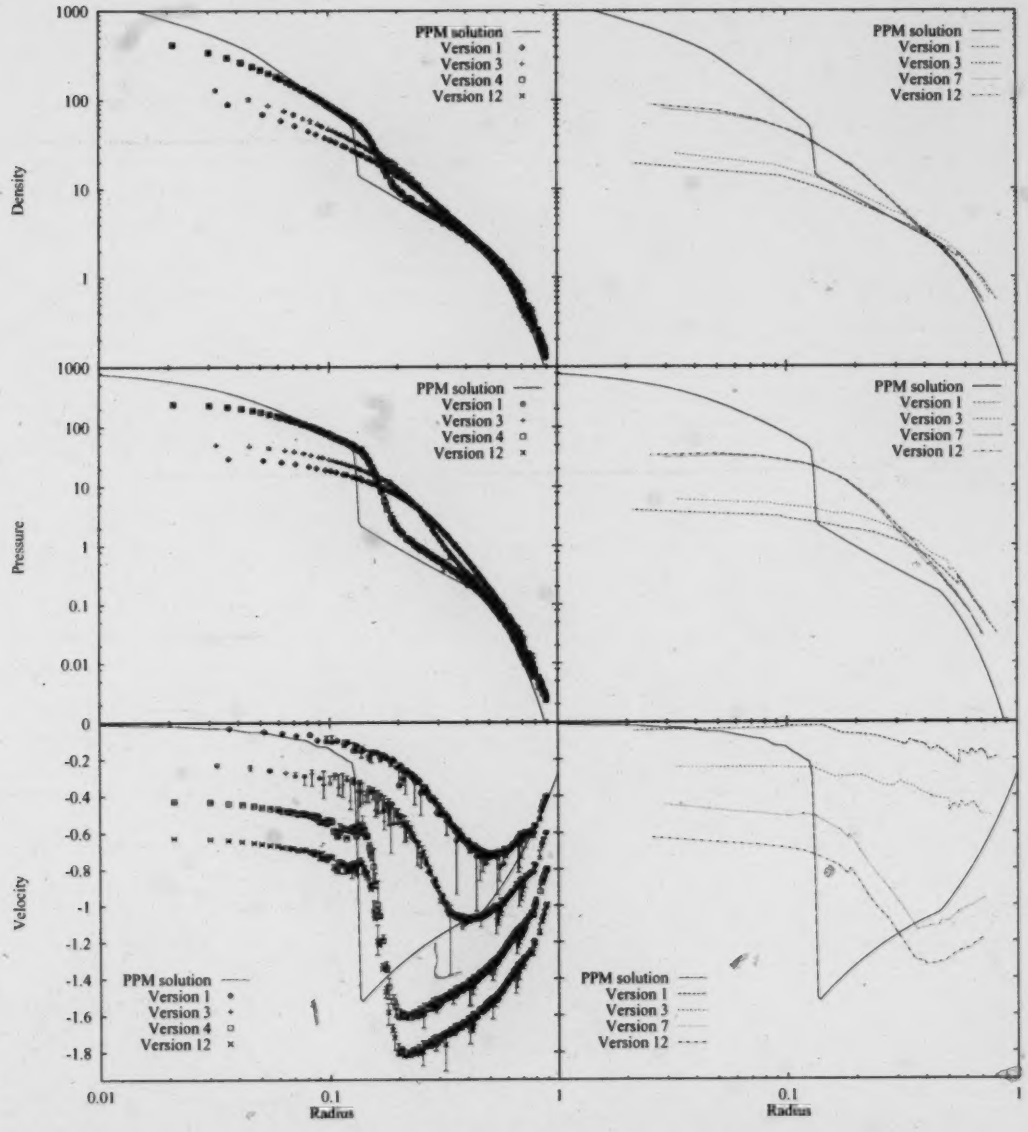

Figure 2.4: Radial profiles for the 30976 and 485 particle collapses at t=0.80 are displayed in the left and right hand columns respectively, with the PPM solution at t=0.77 from SM93 provided for comparison. All values are plotted in normalised units. For the 30976 particle tests the density, pressure and velocity are plotted using Lagrangian bins of size $4 \times 52 = 208$ particles, corresponding to $4\,N_{smooth}$ and for the 485 particle collapses a single line connects all particles. Error bars on the 30976 particle plots show the *maximum* variation within bins. The velocity plots are displaced by intervals of -0.2 vertically for easier interpretation. Notably versions 1 and 3 show significant differences to versions 4, 7 and 12, in particular both the density and pressure are underestimated (in the central regions). Version 3 shows a better capture of the collapsing shock front than version 1, this effect being most evident in the pressure plot. Versions 4, 7 and 12 show excellent agreement, being difficult to distinguish at all radii. Version 7 shows that the inclusion of the shear-free term has little effect on the radial profile.

Table 2.2: Results of Evrard collapse test.

Version	N_{steps}	t_{cpu} (min)	$\lvert\Delta E\rvert/E$ ($\times 10^{-3}$)	$\Delta L/L$ ($\times 10^{-7}$)	N_{par}
1	62	0.21	6.2	11	485
2	62	0.22	3.3	6.7	485
3	66	0.23	1.1	8.2	485
4	94	0.30	2.5	2.4	485
5	88	0.33	1.0	8.4	485
6	129	0.33	1.7	5.0	485
7	104	0.30	0.8	3.2	485
8	95	0.32	0.8	14	485
9	135	0.32	1.2	6.0	485
10	90	0.25	1.5	13	485
11	84	0.20	1.5	8.6	485
12	103	0.20	0.6	6.2	485
1	82	5	6.1	0.6	4776
2	101	6	6.1	1.4	4776
3	98	6	1.9	3.0	4776
4	240	53	3.1	9.7	4776
5	241	61	3.2	5.1	4776
6	231	51	3.5	14	4776
7	243	55	3.2	15	4776
8	252	64	3.1	8.1	4776
9	242	55	3.7	14	4776
10	234	36	2.9	5.8	4776
11	227	18	2.8	6.2	4776
12	241	18	2.8	1.5	4776
1	289	350	0.3	0.3	30976
3	218	321	1.9	0.2	30976
4	418	1319	4.4	0.3	30976
6	411	1262	4.5	1.2	30976
11	405	644	3.6	1.5	30976
12	489	721	4.7	0.6	30976

Values for the 485 particle test are given at t=4.3, for the 4776 test at t=3.4 and for the 30976 test at t=2.0. t_{cpu} is strongly affected (as much as 50%) by system overheads: these values should not be over-interpretated. $\Delta L/L$ is measured relative to the angular momentum the system would have if rotating at the initial circular velocity at R. $|\Delta E|/E$ is the change in the total energy of the system divided by the initial energy, and hence measures the energy conservation exhibited by an implementation. The 485 and 4776 particle tests used a softening length of 0.05R, the 30976 particle tests used a softening length of 0.02R.

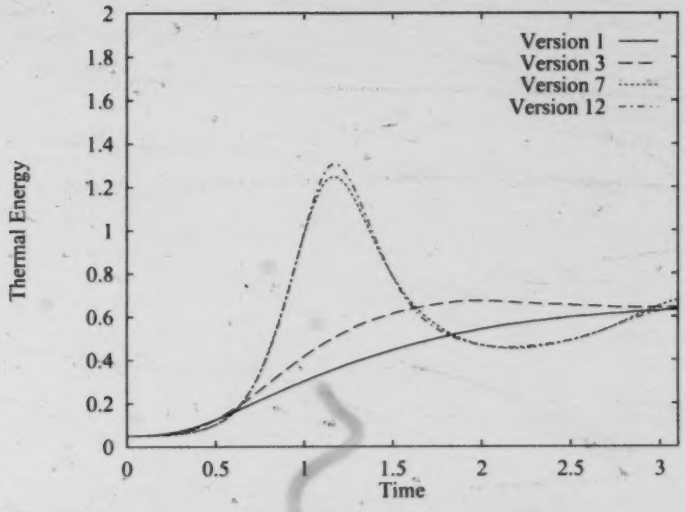

Figure 2.5: Thermal energy plot for the 485 particle collapse, comparing versions 1, 3, 7 and 12. All values are plotted in normalised units. A lower peak thermalization is clearly visible in versions 1 and 3, whilst 7 and 12 show similar profiles. A comparative plot of the energy difference between version 12 and the other codes is shown in figure 2.6.

normalization, $\kappa = 1.5$ versus 1, to run the simulation in a shorter wall-clock time.) The profiles of the Monaghan variants all compare well to the radial PPM solution presented in SM93.

2.7.5 Summary

The relatively poor shock capturing ability of the $\nabla.\mathbf{v}$ viscosity of TC92 is a severe impediment to correctly calculating the evolution of this system. In contrast, the Monaghan viscosities (including the shear-corrected variants) follow the evolution quite accurately. All the Monaghan variants perform well enough in this test to be acceptable algorithms, and when 30976 particles are used it is almost impossible to differentiate between methods (at least in the well resolved core regions). At low resolution (485 particles) the kernel-averaging variants produce a slightly higher thermal peak, although some additional ringing is visible for version 6, but not for 12. At medium resolution (4776 particles) convergence is stronger than the low resolution runs and the difference in post shock ringing is removed. On this basis version 12, when combined with its extremely fast solution time, is the preferable implementation.

2.8 Rotating cloud collapse

A standard test problem for galaxy-formation codes is presented in Navarro and White (1993). In this test a cloud of dark matter and gas is set in solid-body rotation. Gravitational collapse combined with radiative cooling leads to a cool, centrifugally supported gaseous disc.

2.8.1 Initial conditions

The initial radius of the gas cloud is chosen to be 100 kpc, and the total mass (dark matter and gas) is 10^{12} $M_\odot$. The spin parameter, λ, is set to

$$\lambda = \frac{|\mathbf{L}||E|^{1/2}}{GM^{5/2}} \simeq 0.1, \tag{2.34}$$

where $\mathbf{L}$ is the angular momentum, E the binding energy, and M the mass. The baryon fraction is set to 10%, and the cooling function for primordial-abundance gas is interpolated from Sutherland and Dopita (1993). The initial virial ratio for the entire system is $2T/W \simeq 0.08$.

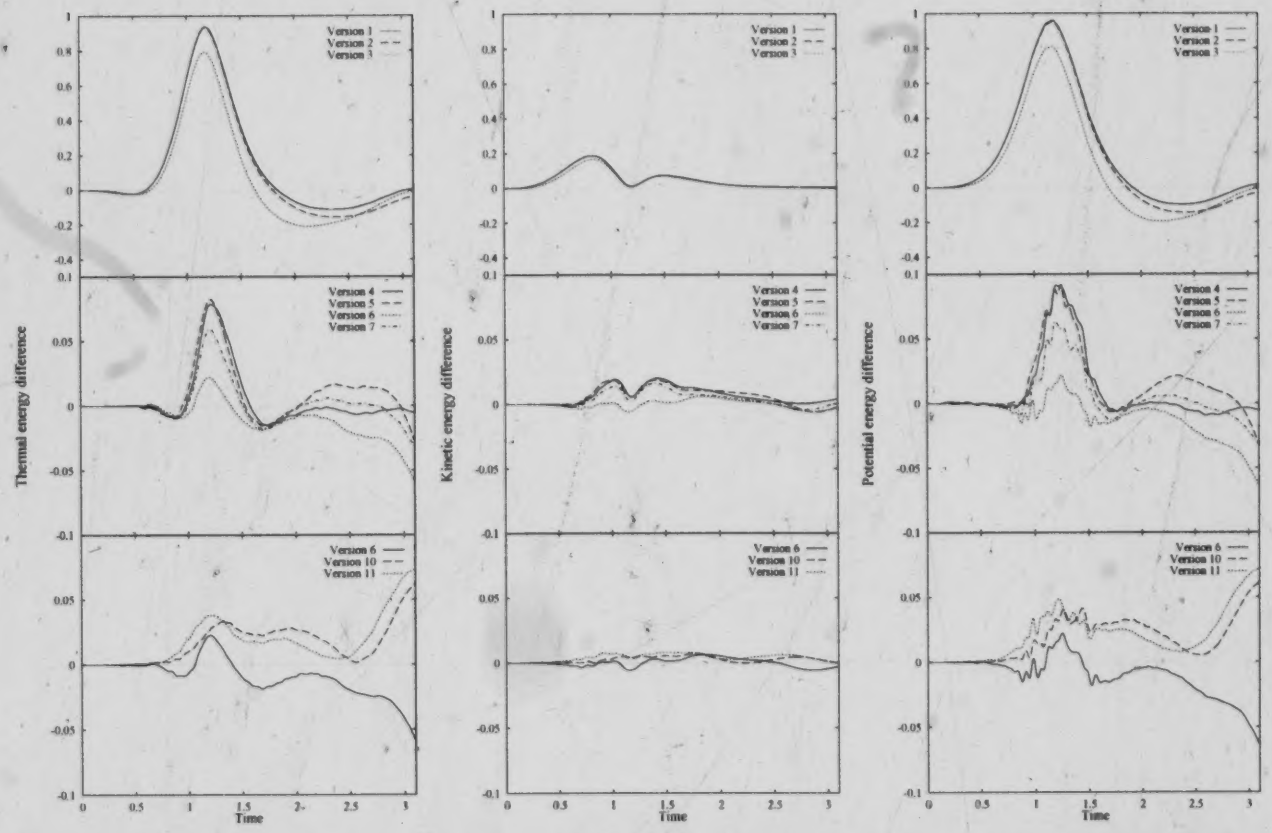

Figure 2.6: N=485 Evrard collapse energy difference profiles. Plots display the energy difference (in normalised units) between version 12 and the version under consideration ($E_{12} - E_{version}$). Column one displays the thermal energy, two the kinetic energy and three the potential energy. The first row of plots compares versions 1, 2 and 3. Note the different energy scaling. The second row, displaying results from versions 4, 5, 6 and 7, shows the effect of the Balsara term (version 7) and also the effect of different symmetrization schemes. The third row compares version 10 and 11, thus indicating the effect of replacing the standard Monaghan viscosity with the single-sided version. Note that the thermal and kinetic energies in the bottom row return close to 0 shortly after t=3–they do not diverge, as suggested by the graphs. Significant differences in performance, particularly for versions 1, 2 and 3, are visible for this particle number.

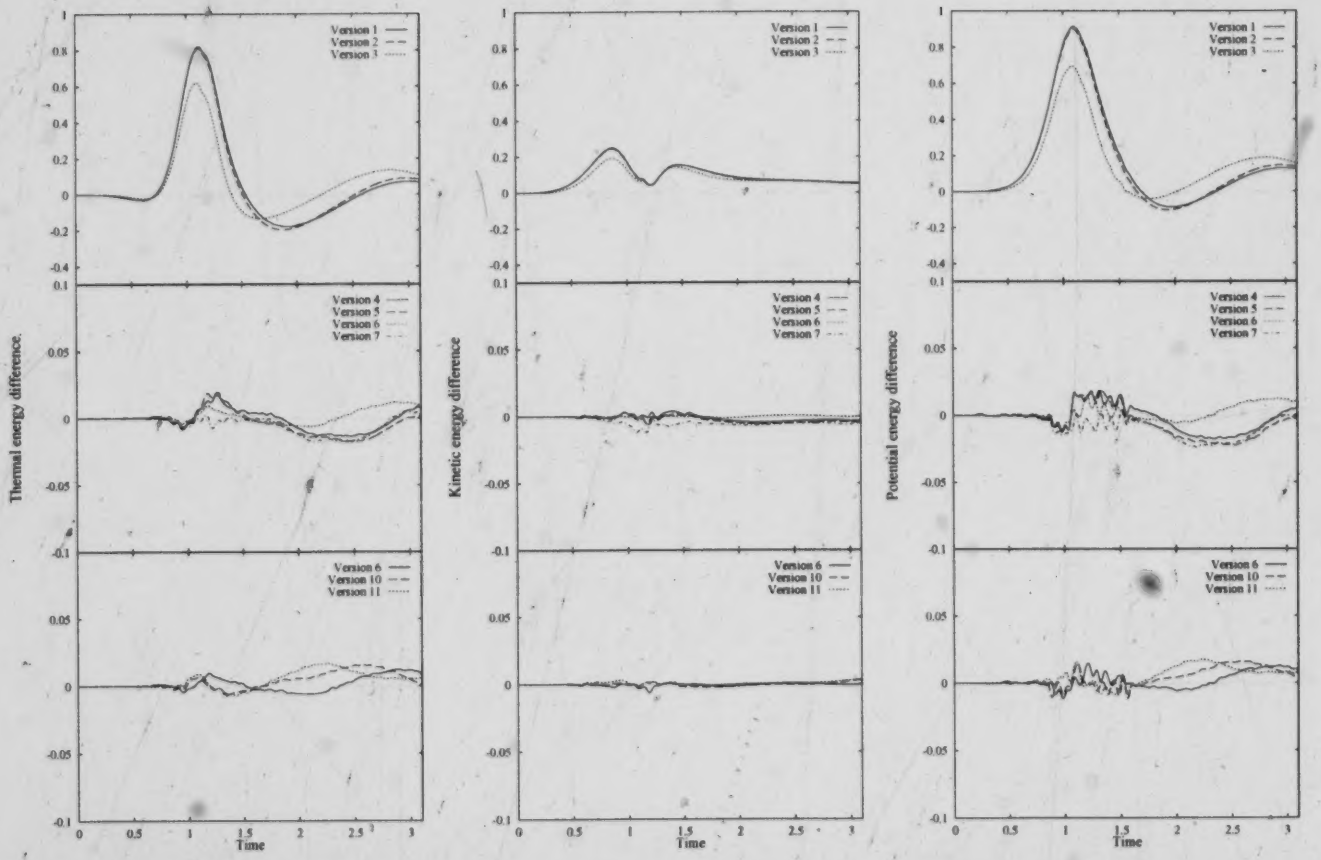

Figure 2.7: N=4776 Evrard collapse energy difference profiles. Plots display the difference (in normalised units) between version 12 and the version under consideration ($E_{12} - E_{version}$). Column one displays the thermal energy, two the kinetic energy and three the potential energy. The first row of plots compares codes 1, 2 and 3. Note the different scaling of the energy. The second row, displaying results from codes 4, 5, 6 and 7, shows the effect of the shear-correction term. The third row demonstrates the effect of replacing the standard Monaghan viscosity (10) with the single-sided version (11). There is stronger convergence to a common solution than shown in the 485 particle tests, with the exception of the kinetic energy for versions 1-3 (the KE being approximately 20% higher than the 485 particle test for version 1). However, the thermal and kinetic energy differerences (which are the dominant contribution) are between 5 to 20% more accurate for versions 1-3.

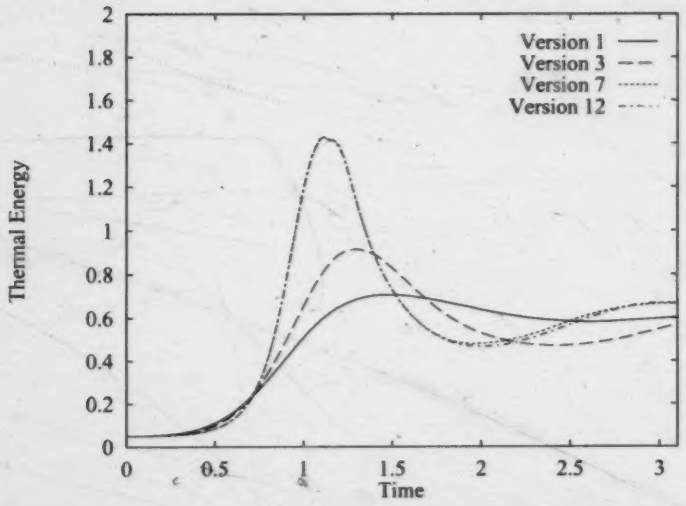

Figure 2.8: Thermal energy plot (in normalised units) for the 4776 particle collapse, comparing versions 1, 3, 7 and 12. As in figure 2.5, versions 1 and 3 exhibit a lower peak thermalization although the peak value is marginally higher. Versions 7 and 12 again have similar profiles and agree very accurately on the peak thermal energy. A comparative plot of the energy difference between version 12 and the other codes is shown in figure 2.7.

For the tests simulations with $N = 2 \times 1736$ particles were performed. The gravitational softening length was set at 2 kpc for both dark matter and gas particles. This is different from previous authors who have set the dark-matter softening to be 5 kpc and the gas softening to 2 kpc. As a result a smaller central dark-matter core is formed. Times are quoted in the units of Navarro and White (1993) (4.7×10^6 yr). One rotation period at the half-mass radius (at t=256) corresponds to approximately 250 time-steps.

The resulting evolution of the system is shown in figure 2.9, and test results summarized in Table 2.3. Radiative cooling during the collapse causes the gas to form a flat disc. The dark matter virialises quickly after collapse, leaving a tight core. Because of the large amount of angular momentum in the initial conditions a 'ring' of dark matter is thrown off. Swing amplification causes transitory spiral features early in the evolution which are later replaced by spiral structure that persists for a number of rotations. If shocked gas is developed during the collapse it forms a halo around the disc.

2.8.2 Non-implementation-specific results

Marginally different results were found for the simulations when compared with other work. This is due to two factors. Firstly, using a 2 kpc softening for the dark-matter particles has a significant effect on the final morphology. The 2-body interaction between gas and dark matter is much stronger than would be expected if the dark matter had a longer softening length. Secondly, most of the particles in the disc have an h value close to h_{min} which in turn sets a significant limit on the minimum mass of a clump that may be resolved. A simulation with $h_{min} = 0.05\epsilon$ was run to observe the effect of this. Figure 2.10 shows a comparison of the simulation run with the smaller h_{min} to the standard $h_{min} = 0.5\epsilon$ simulation. Far more structure is evident on scales close to the gravitational softening length, which must be viewed as being unphysical since at this scale the gravitational forces are severely softened.

Since the circular velocity is calculated from $[GM(< r)/r]^{1/2}$ and the dark matter has the dominant mass contribution, there should be little difference among the rotation curves for the different implementations. Figure 2.12 contains a plot of the rotation curves for four different implementations. Apart from a visibly lower central mass concentration for version 1 there is comparatively little difference.

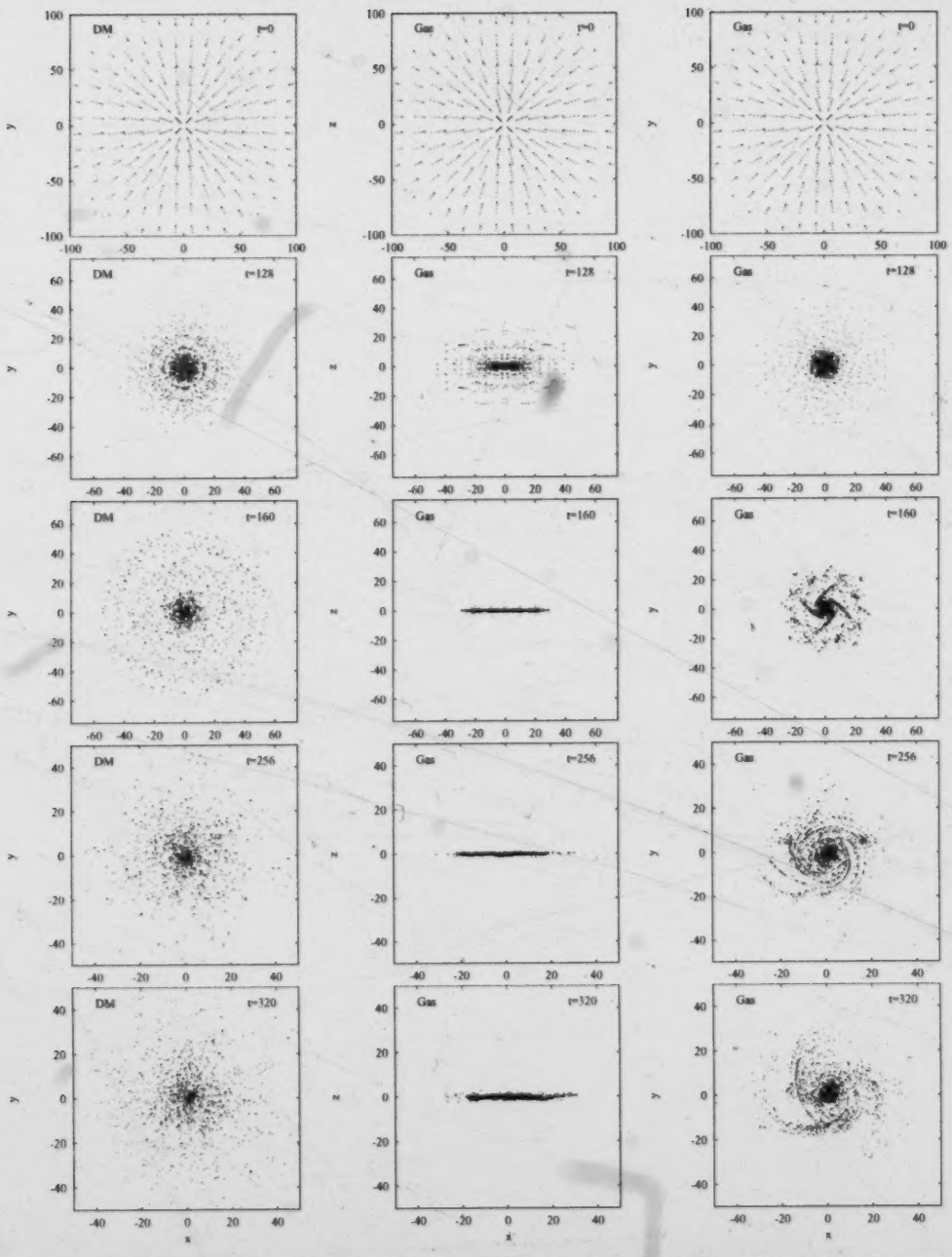

Figure 2.9: Evolution of gas and dark matter in the 2×1736 particle collapse. The results for version 10 are plotted, which produces little shocked gas during collapse. The morphological evolution of the system agrees well with previous work, with minor differences being attributable to differing initial conditions. The results presented here preserve symmetry above and below the equatorial plane for longer than seen in other work, this may be a consequence of the excellent momentum conservation exhibited by grid based gravity solvers and smoothing over all particles within $2h_{min}$.

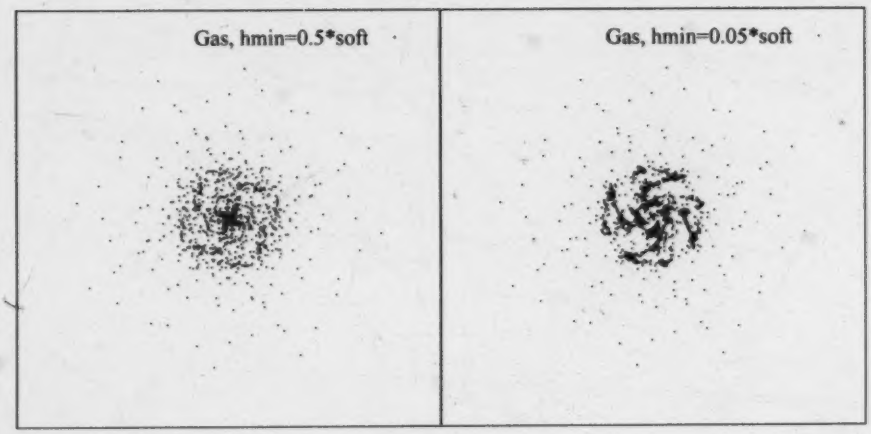

Figure 2.10: Comparison of morphology for the $N = 1736$ run under varying h_{min}, at $t = 128$. Each panel is 100 kpc across, and gravitational softening was set at 2 kpc. Implementation 12 was used to run the simulations. The right-hand panel clearly shows more structure on scales near and below the gravitational softening length.

2.8.3 Implementation-specific results

Before the disc has formed (prior to t=128) versions 1–3 have an extended gas halo compared with the remaining versions. The halo for version 1 is as much as 40% larger than those for versions 4–12. In figure 2.15 the gas structure of version 1 is compared to that of version 12. The source of the extended halo is the $\nabla.\mathbf{v}$ artificial viscosity, which acts to increase the local pressure. This is seen in the early rise in the thermal energy for version 1 in figure 2.13. The more local estimate used in version 3 produces less pressure support and a smaller halo. The Monaghan viscosity does not provide pressure support as the pairwise $\mathbf{r}.\mathbf{v}$ term is very small. The different artificial viscosities also lead to different disc morphologies. The $\nabla.\mathbf{v}$ viscosity fails to damp collapse along the z-axis sufficiently and allows far more interpenetration than the Monaghan viscosity leading to thicker discs in versions 1–3.

The angular momentum losses in Table 2.3 show a noticeable trend. For most codes $\Delta L/L$ is small and negative (indicating a loss of angular momentum). However the shear-corrected Monaghan variants show an *increase* in the angular momentum of the system. However, since the magnitude of the angular momentum is approximately the same as that of the other codes, there is no strong significance to this result. Version 2 also has the shear-correction term, but the similar performance to version 1 can be attributed to the low amount of dissipation produced by the $\nabla.\mathbf{v}$ viscosity.

Examining the thermal energy during collapse yields very interesting results. Figure 2.13 shows a plot of the thermal energy of the cloud versus time. As a fraction of the total energy the thermal energy makes a small contribution because the baryon fraction is only 10%. However, the relative differences in thermal energy between versions can be significant. This situation is analogous to the differences seen in the kinetic energy in the Evrard collapse test (see section 2.7). For this test it is important to note that the differences in the thermal energy arise from the amount of shocked gas present in the simulation, which is determined by the artificial viscosity. This is demonstrated in the comparison plot of the fraction of gas above 30,000 K, shown in figure 2.14. A small change in the artificial viscosity can have a very significant change in the amount of shocked gas. Changing the h-averaging scheme from the arithmetic mean to the harmonic mean, will lead to the artificial viscosity being stronger in most situations. This is because the term used to prevent divergences, $0.01\bar{h}_{ij}^2$ in the denominator of equation 2.27, is now *always* smaller, and hence can lead to larger values of Π_{ij}. This explains why version 5 has so much shocked gas. Similarly, the ρ_{ij} replacement in versions 11 and 12 leads to a larger Π_{ij} as the $(h_i/h_j)^3\rho_i$ replacement systematically tends to underestimate the value of ρ_j, and hence more shocked gas is produced. Note, however, that the effect is only noticeable in cases of extreme density contrast, *e.g.* halo particles just above a cold

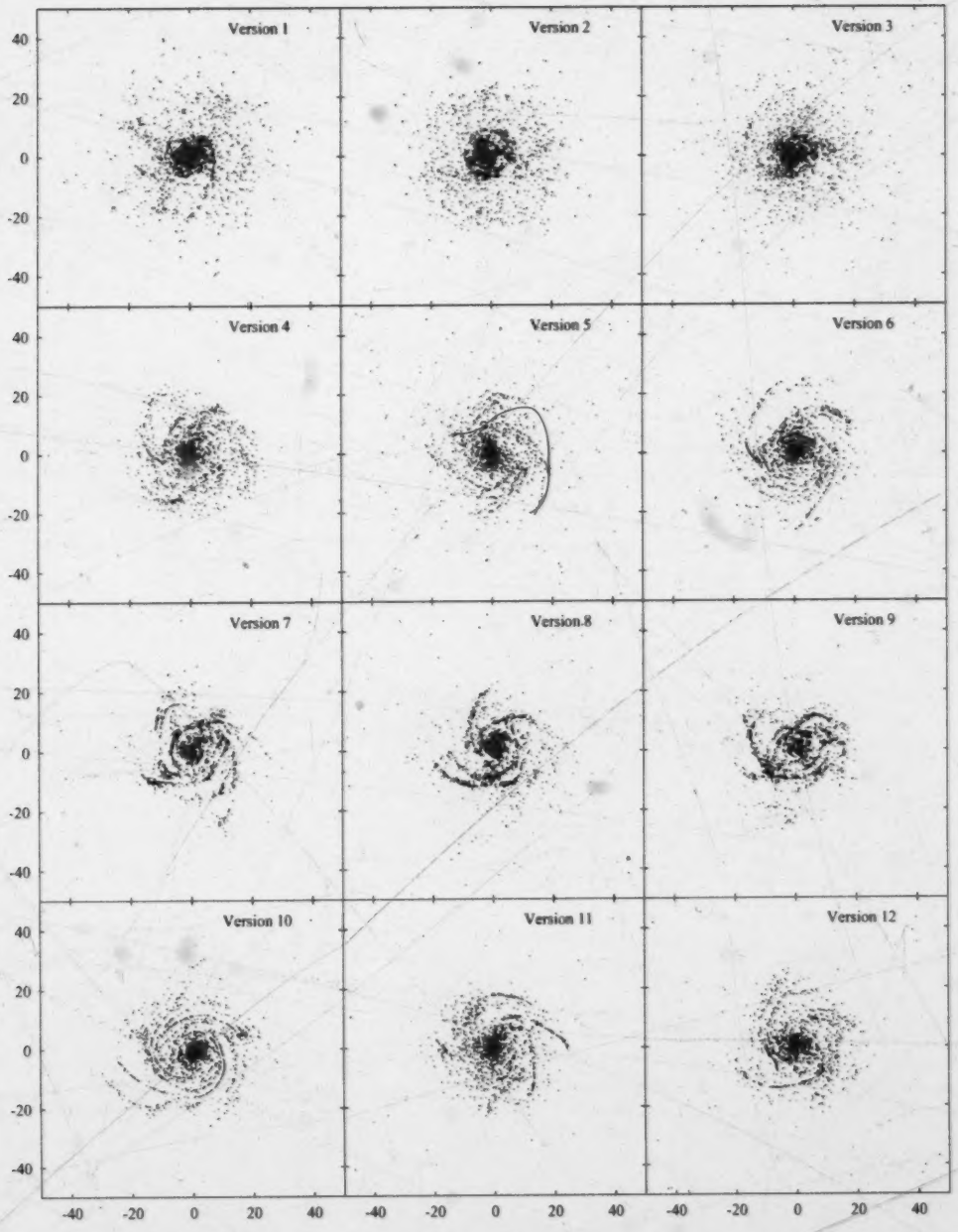

Figure 2.11: Comparison of gas morphology for 2 × 1736 particle collapse. Time is t=256 and axis scales are in kpc. Clearly different implementations exhibit different spiral structures, indicating that at this resolution the structures are poorly defined.

Table 2.3: Results of rotating cloud collapse test.

Ver	N_{step}	$\|\Delta E\|/E(\times 10^{-3})$	$\Delta L/L(\times 10^{-4})$	Q_{peak}	f_{peak}
1	1461	2.8	−0.23	0.001	0.095
2	1627	1.7	−0.98	0.001	0.118
3	1588	1.7	−3.8	0.024	0.212
4	1806	2.8	−9.9	0.023	0.159
5	1703	1.8	−9.2	0.049	0.215
6	1834	1.8	−10.	0.009	0.124
7	2087	4.4	+6.2	0.015	0.128
8	2002	4.4	+6.9	0.030	0.164
9	2074	4.8	+5.7	0.004	0.095
10	1789	2.2	−10.	0.002	0.074
11	1748	3.1	−7.4	0.023	0.143
12	1705	2.9	−6.6	0.036	0.175

With the exception of the peak variables values are given at t=256. $\Delta L/L$, the fractional change in angular momentum, is expected to be zero. Q_{peak} is the maximum value of the thermal energy (as a fraction of the initial total mechanical energy) and f_{peak} is the peak fraction of gas shocked to high temperature.

gaseous disc.

It is also evident that a change in the symmetrization procedure can have a significant effect. Codes 4 and 6 have differing amounts of shocked gas. This fact suggests that the halo gas in this collapse problem must sit at the edge of the Rees-Ostriker (1977) cooling criterion, namely that the free fall time is approximately equal to the gas cooling time. This was checked by running simulations with masses a factor of five higher and two lower. The lower mass system produces less (12% by mass, compared to 18%) shocked gas, whilst the high mass system produces a very large (75%) amount of shocked gas. In view of these results, and that the thermal energy is a very small fraction of the total, strong emphasis should not be placed on the differences in shocked gas between the versions. Comparison of the results of Serna *et al.* (1996) and those of Navarro and White (1993) confirms this conclusion as the amount of shocked gas in the former differs visibly from the latter. Correctly calculating the evolution of the halo gas is very difficult given the poor resolution of the density gradient, and hence cooling, combined with the sharp nature of the Rees-Ostriker Criterion (the cooling curve exhibits order of magnitude changes dependent upon temperature). These results mirror those of section 2.7 where although evolution varied for different algorithms, the final state of the system was very similar.

2.8.4 Summary

Setting aside the differences in the amount of shocked gas among implementations, there is comparatively little variation among the final results. There are differences in morphology: versions 1, 2 and 3 have a thicker disc and, during the initial collapse, show a more extended gas halo. Both effects can be traced to the $\nabla.\mathbf{v}$ viscosity. The potential and kinetic energies during collapse, however, are all similar, due to the dominance of the dark matter.

2.9 Angular momentum transport

Disc stability is a critical issue in galaxy formation. It is now widely known (Balsara, 1995; Navarro and Steinmetz, 1997, for example) that the standard Monaghan viscosity introduces spurious angular momentum transport (as opposed to the physical transfer of angular momentum that occurs in differentially rotating discs). This spurious transfer can have a significant effect on the development of a small N object, as the angular momentum may be transported to the outer edge of the disc in a few rotations (Navarro and Steinmetz, 1997).

This section compares the growth of the half-angular-momentum radius (half-AM radius) of the disc, the half-mass radius and ratio of the two to determine which SPH implementation is least susceptible to this problem. A similar investigation was first performed by Navarro and Steinmetz

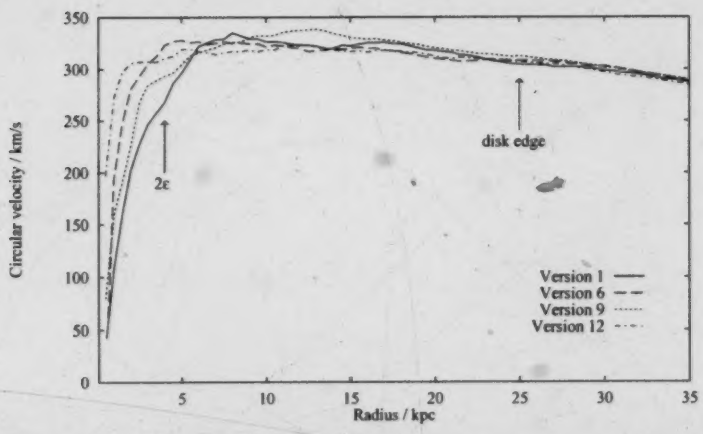

Figure 2.12: Rotation curves for four different implementations. The shear correction in version 9 produces a higher rotational velocity at half the disc radius. This is because the outward transport of angular momentum is reduced, thereby reducing the inward movement of the half mass radius.

(1997). The initial conditions were provided by the disc formed by version 9 during the rotating cloud collapse (see section 2.8). The central 100 kpc diameter region was cut out and evolved for a sufficient time to allow relaxation of the system as well as for transport of angular momentum. Higher resolution tests were not conducted as increasing the resolution leads to a disc that is unstable to perturbations (Navarro and White, 1993).

2.9.1 General evolution properties

Given the comparatively quiet disc environment the morphological evolution of the different versions were comparatively similar. The only noticeable difference could be seen in the disc thickness, which for the $\nabla.\mathbf{v}$ viscosity variants (1, 2 and 3), was much larger than that of the rest of the versions. This effect was also observed in section 2.8, although here the gas has 'diffused' away from an initially thin disc.

2.9.2 Implementation-specific results

The evolution of the half-AM radii are plotted in figure 2.16. The figure demonstrates that the codes fall into two groups, with one group suffering a stronger decay of the half-mass radius and an associated growth of the half-AM radius. The other group, which shows less decay, consists of versions 1–3, 7–9. The artificial viscosities for this group are the $\nabla.\mathbf{v}$ variants (1–3) and the shear-corrected Monaghan version (7–9). For this group the half-AM radius does not change significantly during the simulation which corresponds to approximately 20 rotations (at the *initial* half-mass radius) and 5000 time-steps. The half-mass radius decays by approximately 50%, leading to a similar reduction in the half-mass to half-AM radii ratio.

Within this first group of codes there is a sub-division determined by the disc thickness at the end of the simulation. The $\nabla.\mathbf{v}$ variants all have a thick disc, due to the failure of the algorithm to adequately damp convergent motions in the z-direction. For the shear-corrected Monaghan variants this is not a concern.

The second group, comprising versions 4–6, 10–12 (all Monaghan viscosity variants), shows significant outward transport of the angular momentum, and by the end of the simulation the half radius has increased by approximately 50%. There is also a larger decay in the half-mass radius, it being 50% greater than the decay seen for the other group of codes. These two results contribute to make the half-mass to half-AM radii ratio decay to only 25% of its initial value at the beginning of the simulation. This result confirms that seen in Navarro and Steinmetz (1997) where it was shown that the shear correction significantly reduces angular momentum transport in disc simulations. The results presented here show a faster increase in transport, but this is probably due to the simulations being dissimilar; Navarro and Steinmetz place a disc in a predefined halo, and then evolve the combined system.

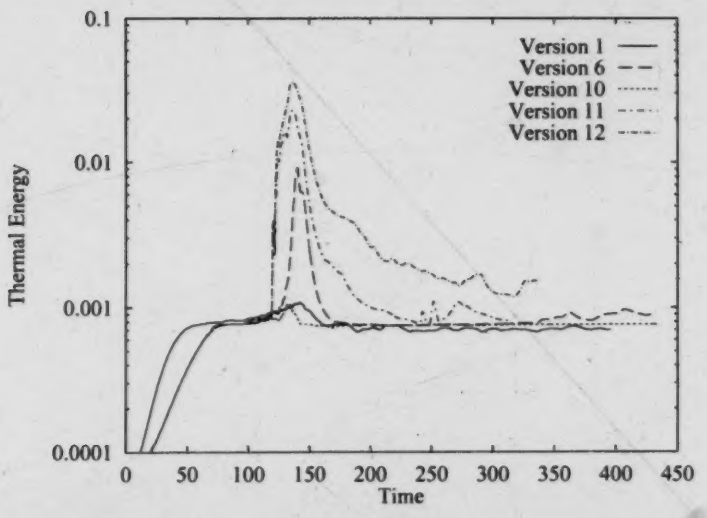

Figure 2.13: Thermal energy, divided by the total initial mechanical energy, for 5 different implementations. A marginal change in the artificial viscosity can produce a 25-fold change in the peak total thermal energy. This is indicative of the test sitting at the edge of the Rees-Ostriker (1977) criterion.

2.9.3 Summary

Whilst preserving the half-mass to half-AM radii ratio as well as the shear-corrected scheme, the $\nabla.\mathbf{v}$ viscosity is not a significant improvement. The large increase in disc thickness and loss of definition more than outweighs the improvement in the decay of the half-mass radius. Both the Monaghan and shear-corrected artificial viscosities maintain the disc structure. On the basis of these tests any of the shear-corrected implementations (7–9) is to be preferred.

2.10 Cosmological simulation

An important use of simulations in cosmology is to model the formation and evolution of galaxy halos on large scales and to provide statistical descriptions of their distribution. Whilst investigations of this type have frequently been undertaken in collisionless simulations, with dark matter halos being identified as the sites of galaxy formation, the trend is towards more realistic descriptions of the process of baryonic condensation within dark matter halos. A significant problem, as yet unresolved, is the increased cooling of baryons in dark matter halos as the resolution in simulations is increased: the "cooling catastrophe". Higher resolution leads to earlier formation of structure at higher densities with enhanced cooling and containing a greater fraction of the baryons. Physically this problem is believed to be alleviated by feedback from stars which heats the gas and regulates the transfer of gas into the cold phase. In the model described here, these processes are crudely mimicked in an attempt to avoid enhanced cooling, by virtue of the high particle mass as described below.

This test is a simple example of the formation and distribution of cold, dense gas within a cosmological volume. The focus is to investigate the sensitivity of the results to the particular choice of SPH implementation. Recovering consistent positions and masses for the objects is particularly important. The resolution is such, however, that no internal information (such as spiral structure or radial density profiles) can be recovered.

2.10.1 Initial Conditions

The simulations presented here were of an $\Omega_0 = 1$, standard cold dark matter universe with a box size of $10h^{-1}\,\mathrm{Mpc}$. $h = 0.5$ is set throughout this section, equivalent to a Hubble constant of $50\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$. The baryon fraction, Ω_b was determined from nucleosynthesis constraints, $\Omega_b h^2 = 0.015$ (Copi *et al.*, 1995), and a constant gas metallicity of $0.5Z_\odot$ was used. Identical initial

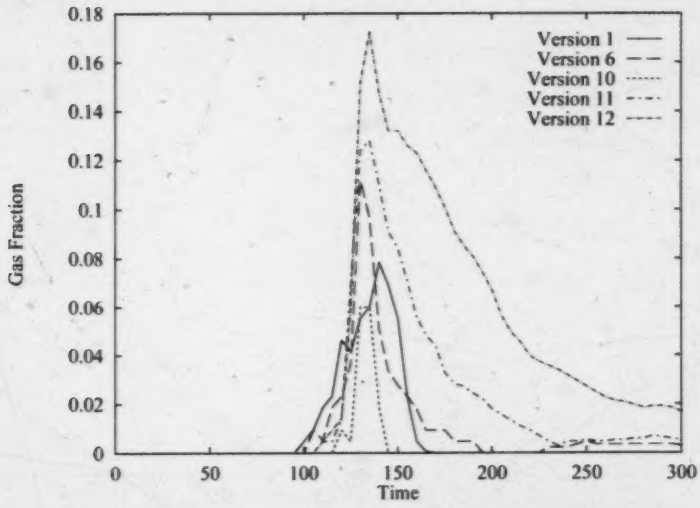

Figure 2.14: Fraction of gas mass above 30,000 K. This gives an approximate measure of the amount of gas which is shocked. The amount of shocked gas is extremely sensitive to the artificial viscosity implementation. Compare versions 10 and 11 which differ only by the $\bar{\rho}_{ij}$ substitution, the peak amount of shocked gas is different by a factor of two.

conditions were used in all cases, allowing a direct comparison to be made between the objects formed.

The initial fluctuation amplitude was set by requiring that the model produce the same number-density of rich clusters as observed today. To achieve this we take $\sigma_8 = 0.6$, the present-day linear rms fluctuation on a scale of $8h^{-1}$ Mpc (Eke *et al.*, 1996; Viana and Liddle, 1996). Each model began with 32^3 dark matter particles each of mass $1.58 \times 10^{10}\,M_\odot$ and 32^3 gas particles each of mass $1.01 \times 10^9\,M_\odot$, smaller than the critical mass derived by Steinmetz and White (1997) required to prevent 2-body heating of the gaseous component by the heavier dark matter particles. The simulations were started at redshift $z = 19$ when the gas is assumed to have a temperature of $10^4 K$. The simulations employ a comoving Plummer softening of $10h^{-1}$ kpc, which is typical for modern cosmological simulations but still larger than required to accurately simulate the dynamics of galaxies in dense environments. The cooling function is interpolated from the data in Sutherland and Dopita (1993). This test case is identical to that extensively studied by Kay *et al.* (1999) who used it to examine the effect of changing numerical and physical parameters for a fixed SPH implementation.

2.10.2 Extraction of glob properties

The gas is effectively in three disjoint phases. There is a cold, diffuse phase which occupies the dark-matter voids and therefore most of the volume. A hot phase occupies the dark-matter halos and at this resolution is typically above 10^5 K. Finally, there is a cold, dense phase consisting of tight knots of gas typically at densities several thousand times the mean and at temperatures close to 10^4 K. The relative proportions of the gas in each of these phases is given in Table 2.4. The fraction of the gas which resides in the hot phase is nearly constant across all the runs. This fraction is largely determined by the depth of the dark matter halos which remain nearly invariant, each of which contains gas near to virial equilibrium. As gas cools, pressure support is removed from the middle of the halo and more of the void material collapses into the halo and heats up. Following Evrard, Summers and Davis (1994), a cooled knot of particles is known as a 'glob' because the resolution is such that they can hardly be termed a galaxy. The properties of the globs are calculated by first extracting all the particles which are simultaneously below a temperature of 10^5 K and at densities above 180 times the mean and then running a friend-of-friends group-finder with a maximum linking length, b, of 0.08 times the mean interparticle separation of the dark matter. In practice the object set obtained is relatively insensitive to the choice of b because the globs are typically disjoint, tightly bound clumps. The cumulative multiplicity function for the different implementations is shown in

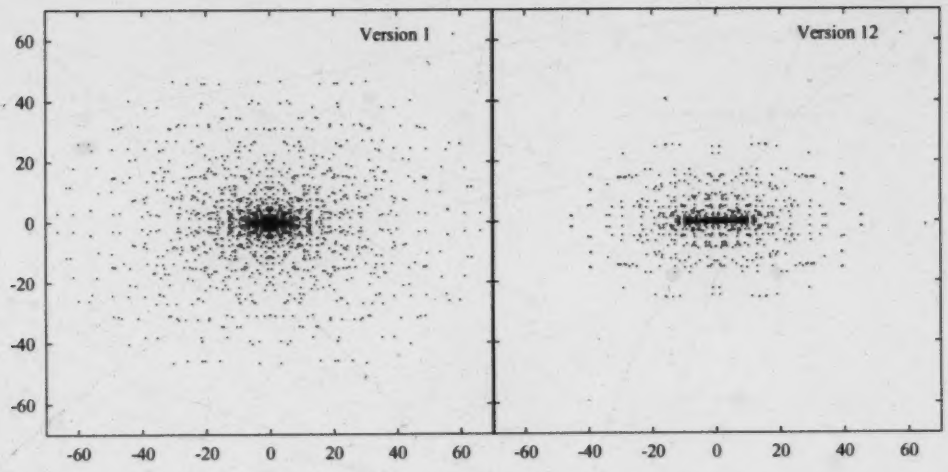

Figure 2.15: Comparison of gas halo size at $t = 128$ for versions 1 and 12. Axis scales are given in kpc. The gas halo for version 1 is clearly larger.

figure 2.17.

2.10.3 Results of cosmological test

In all cases the largest object has 'overcooled' in the sense described in section 2.6. It is much too massive to be expected in a simulation of this size and is only present because gas within the hot halo has its cooling rate enhanced by the very high-density gas contained in the globs.

A distinct difference can be seen in the morphologies of small objects formed by versions 1–3 compared to those formed by 4–12. Versions 1–3 produce spherical objects since the $\nabla.\mathbf{v}$ viscosity used does not damp random orbital motion within the softening radius. Versions 4–12 produce disc-like objects as a result of the effective dissipation provided by the pairwise trigger and conservation of angular momentum. Both spherical and disc objects are of a size approximately equal to the softening length. Merging is not expected to play a significant role in this simulation because at the given resolution limit it is only possible to resolve large galaxies and halos and the simulation volume contains only a small number of these objects.

The major discriminant between the versions is the different artificial viscosities. As shown in section 2.4, versions 1, 2 and 3 produce broader shock fronts because the viscosity employed is an averaged quantity, whereas for all the other versions the viscosity is calculated on a pairwise basis. The fraction of matter present in globs and the number of groups with $N > 50$ is clearly lower for versions 1–3 than for other versions. The lower mass-fraction in globs is a combination of both the smaller number of groups found above the threshold and versions 1–3 producing lighter objects. As was demonstrated in section 2.7, the $\nabla.\mathbf{v}$ viscosity produces a shallow collapse, with much less dissipation. In these versions, collapsed objects containing approximately N_{smooth} particles will experience virtually no shock heating with the gas particles simply free-streaming within the shallow potential wells. In simulations with cooling the $\nabla.\mathbf{v}$ viscosity provides a marginally higher pressure support than the Monaghan viscosity (see section 2.8) which can be sufficient to prevent collapse of surrounding material. At low resolution this results in an object not achieving $N > 50$, whilst at higher resolution the object has lower mass. Thus 'halo' gas that is found in f_{gal} for the Monaghan variants is found in f_{cold} for the $\nabla.\mathbf{v}$ variants.

Of the Monaghan variants 4–6, version 5 has the lowest fraction of matter in the glob phase and the highest fraction of hot gas. Fig. 2.18 shows that it also produces systematically lighter objects. Version 6 has the highest fraction of matter in the glob phase, the lowest fraction of hot gas and tends to produce the heaviest objects. These results are due to the symmetrization scheme causing the artificial viscosity to produce different amounts of dissipation. Similarly, for versions 10–12 the fraction of hot gas can be traced to the amount of dissipation. These results are consistent with those in section 2.8, where they are discussed in detail. The trend for Monaghan-type viscosities is

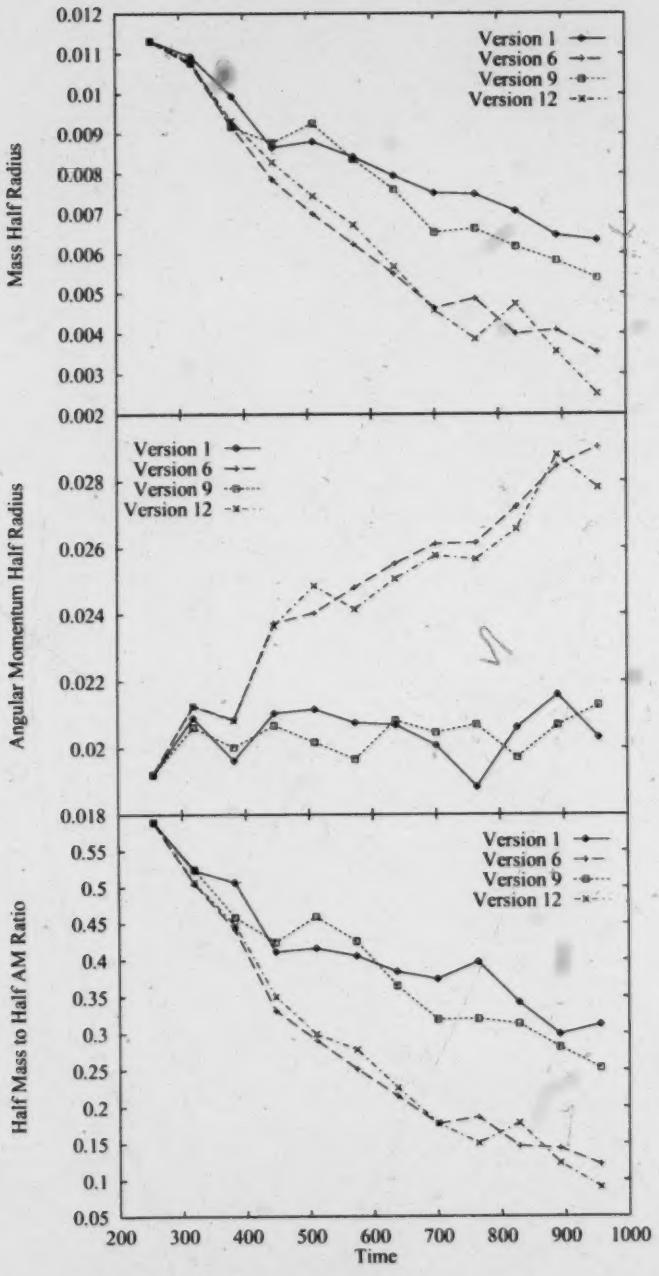

Figure 2.16: The half-mass radius, half-AM radius and the ratio of the two. Versions 1 and 9 are best, producing a negligible increase in the angular momentum half-mass radius and a comparatively slow reduction in the half-mass radius. The Monaghan variants that are not shear-corrected perform worst, transporting the angular momentum rapidly, resulting in a fast decrease of the half-mass radius. Radii are plotted in code units (1 unit equals 800 kpc).

Table 2.4: Properties of the cosmological test runs by version.

Version	N_{steps}	Hours	$N_{>50}$	M_{big}	f_{gal}	f_{hot}	f_{cold}
1	6060	34.7	30	1.01	0.17	0.37	0.45
2	6348	35.0	34	1.07	0.19	0.36	0.45
3	6532	43.7	41	1.08	0.20	0.34	0.46
4	6984	65.1	49	2.41	0.30	0.37	0.33
5	6113	60.5	48	2.34	0.27	0.40	0.33
6	6769	57.4	56	2.53	0.34	0.34	0.33
7	7562	63.4	47	2.43	0.29	0.34	0.36
8	6782	93.6	47	2.38	0.27	0.36	0.36
9	7688	109.7	49	2.40	0.32	0.37	0.31
10	6858	60.5	50	2.69	0.33	0.33	0.34
11	6522	40.4	48	2.43	0.31	0.35	0.34
12	6205	38.5	51	2.28	0.30	0.35	0.34

Listed are the number of steps taken to reach $z = 0$, the number of hours required on a Sun Ultra II 300 workstation, the number of groups of more than 50 cold particles found at the endpoint, the mass of the largest clump (in units of $10^{12}\,M_{\odot}$), the fraction of the gas in galaxies, the fraction of gas above 10^5 K and the fraction of the gas that remains diffuse and cold (all at $z = 0$).

distinct: versions that produce more dissipation form lighter objects as the hot halo gas is heated to higher temperatures where the cooling time is longer.

In section 2.4 it was shown that the shear-correction term is less able to capture shocks and consequently produces lower shock heating. For the h-averaging implementations (4, 5), the hot gas fraction is reduced upon adding shear correction, which agrees with the shock tube result. For the kernel-averaged version this is not the case – the hot gas fraction increases. This result is probably not significant; in section 2.8 versions 4–6 all show reduced dissipation upon including the shear-correction term.

2.10.4 Summary

In principle any of the versions discussed in this paper could be used effectively for this problem. The masses of the objects formed vary by as much as 50% but this is due to the different effective resolutions introduced by the various artificial viscosity prescriptions and the fact that increasing the resolution leads to more cold material (the cooling catastrophe). These simulations were performed in a region of parameter space where the amount of cooling gas has not yet converged, and so such differences were to be expected.

Any of the prescriptions could be used since only the position and mass of the galaxies formed are of interest–not differences in their internal structure below the gravitational softening scale. Object-by-object comparisons between the runs display a good positional match, as shown in figure 2.18. Although globally more gas cools for the runs with a Monaghan type viscosity all the individual object masses scale by close to the same factor. Cosmetically, versions 4–12 of the code produce more visually appealing outputs and disks are perhaps to be physically preferred to amorphous blobs but it should be stressed that these differences occur on scales below the gravitational softening and so are of limited significance.

2.11 Conclusions

A series of tests has been presented which are designed to determine the differences in performance of various SPH implementations in scenarios common in simulations of cosmological hierarchical clustering. Special attention was paid to how the codes perform for small-N problems. A summary of the findings is presented in Table 2.5.

Principle conclusions follow.

1. Schemes that use the Monaghan viscosity supplemented with the shear-correction are recommended. Of those methods that do not use the shear-correction, version 12 is preferable because of its high speed and accuracy (and see 7 below).

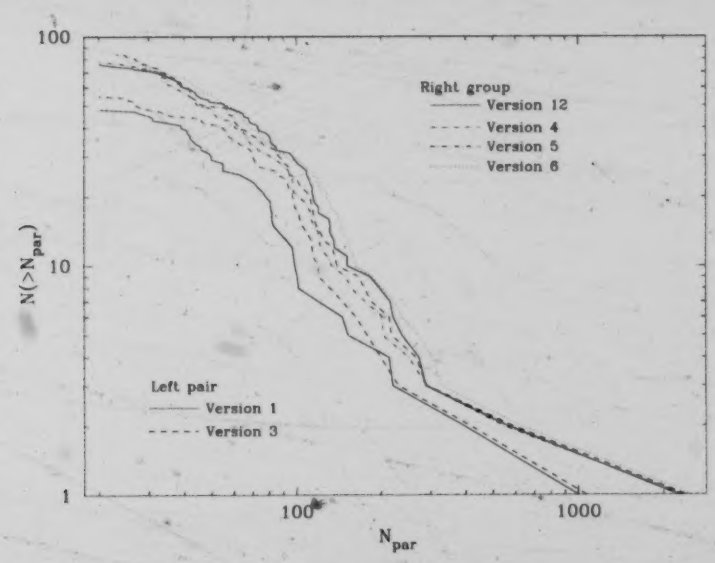

Figure 2.17: The multiplicity functions for different versions. The six implementations plotted span the range of outcomes as, in all cases, the addition of a shear-correction to the viscosity makes little difference. Versions 10 and 11 produce very similar results to version 12, whilst 1, 2 and 3 produce a smaller number objects than the other versions.

2. Several implementations introduce programming difficulties, such as changing the neighbour search from a gather process to a hybrid gather-scatter. This is especially problematic in the adaptive refinements where different symmetrization schemes lead to different choices for which particles to place in a refinement. The TC92 symmetrization is by far the easiest to program in this respect.

3. The choice of artificial viscosity is the primary factor in determining code performance. The equation of motion and particle symmetrization schemes have only a secondary role, albeit significant for small-N. In particular, the artificial viscosity used in the current implementation of HYDRA , and variants of it, produce a large amount of scatter in local variables, such as the velocity field and temperature, and also lead to less thermalisation. These characteristics indicate that the relative performance of an artificial viscosity is determined by its effective resolution. Viscosities which use an estimate of $\nabla.\mathbf{v}$ to determine whether the viscosity should be applied will always be less able to capture strong flow convergence and shocks than those which use the pairwise $\mathbf{r}.\mathbf{v}$ trigger.

4. Instabilities inherent in simple smoothing-length update algorithms can be removed. By using a 'neighbour-counting kernel' and a weighted update average, stability can be increased without requiring an expensive exact calculation of the correct h to yield a constant number of neighbours.

5. The results agree with the conclusion presented in SM93 that to accurately calculate local physical variables in dynamically evolving systems, at least 10^4 particles are required. The difference in morphologies observed in the rotating cloud collapse clearly indicates that the belief SPH can accurately predict galaxy morphologies with as few as 1000 particles is overly optimistic. This implies that in studies of galaxy formation, where the internal dynamics define morphology, an object may only be considered well resolved if it contains a minimum of 10^4 particles.

6. The introduction of a shear-corrected viscosity leads to reduced shock capturing, although the effect is small and primarily visible in the local velocity field. The results of other authors were confirmed, namely that the shear-corrected viscosity does indeed reduce viscous transport of angular momentum.

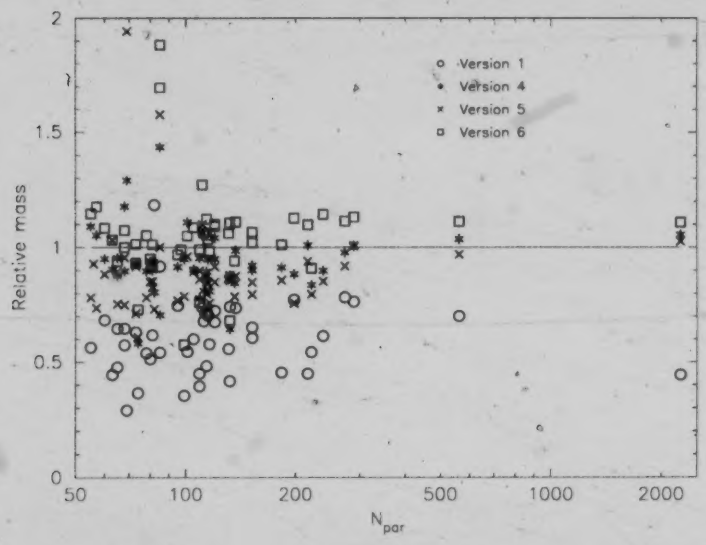

Figure 2.18: Comparison of object masses in different implementations. The masses of objects found by the group-finder in version 12 of the code are compared to the masses found for the corresponding objects in other versions. Versions 1–3 of the code all produce objects of about half the mass and many smaller objects are missing (because in these runs they fall below the resolution limit of around 50 particles). As for figure 2.17 the addition of a shear-correction makes little difference and versions 10 and 11 of the code produce very similar masses to version 12.

7. It is possible to implement a scheme (11 and 12) which uses the Monaghan artificial viscosity, that does not require the precomputation of all density values before solving the hydrodynamic equation of motion. Further, the resulting implementation conserves momentum and energy to an extremely high accuracy. This implementation removes the need to compute the list of neighbours twice or alternatively the need to store it for every particle. Hence it is significantly faster than other schemes. Additionally, although not done in this work, the shear-correction can be incorporated with little extra effort.

Thus, on the basis of these conclusions, the following chapters use version 12, supplemented with the shear-correction algorithm.

Version	Shock capturing 3.1, 3.2	Cooling 3.3	AM loss 3.7	Morphology 3.6 (3.5, 3.7)	Drag 3.4	Execution time
1	×	√	√	×	√	1.1
2	×	√−	√	×	√	1.1
3	×	√	√	×	√	1.1
4	√	√	×	√	√	2.4
5	√	√	×	√	√	2.4
6	√	√	×	√	√−	2.4
7	√−	√	√	√	√+	2.4
8	√−	√	√	√	√+	2.4
9	√−	√	√	√	√	2.4
10	√	√−	×	√	×	1.2
11	√	×	×	√	×	1.1
12	√	√−	×	√	√	1.1

Table 2.5: Qualitative summary of the strengths and weaknesses of each implementation. Each version is catagorized using a √ to indicate preferable performance and an × to indicate inferior performance. + and - signs differentiate between similarly grouped implementations. Assessments in the morphology column refer to the success with which the implementation can produce and maintain thin discs. Execution times represent the best performance to be expected from each algorithm and are relative to the HYDRA code without the new h-update algorithm. In high-resolution simulations (2×128^3 particles, for example) versions 4–9 may well be significantly slower. Section numbers from which these conclusions are drawn are shown at the top of the columns.

Chapter 3

Incorporating star formation and feedback into simulations of galaxy formation

"We think in generalities, but we live in details."

–Alfred North Whitehead

3.1 Introduction

A detailed understanding of galaxy formation in hierarchically clustering universes remains one of the primary goals of modern cosmology. Unlike the study of gravitational clustering of dark matter, a number of complex physical processes contribute to galaxy formation. Further complication is evident in that a significant number of these processes cannot be modelled from first principles, the main example of this being star formation.

Analytic and semi-analytic theory of galaxy formation is a well-developed field (see White, 1994, for an overview). The theoretical framework for studying the condensation of baryons in dark matter halos was laid out by White and Rees (1978) following the foundation work on hierarchical clustering by Peebles (1980, and references therein). White and Rees illuminated the fact that for baryons condensing in dark matter halos the cooling time of the gas is always shorter than the free-fall collapse time. In a related paper, Fall and Efstathiou (1980) demonstrated that disc galaxies could be formed in a hierarchical clustering model, provided that the gas maintains its angular momentum. Later work by White and Frenk (1991) further developed this hierarchical clustering model and identified the *cooling catastrophe* for CDM cosmologies. The cooling catastrophe is caused by the large mass fraction in small high-density halos at early times. The gas resident in these halos cools on the time-scale of a Myr, thus precipitating massive star formation very early on in the development of the CDM cosmology (at odds with the observations of our Universe). To circumvent this problem White and Frenk introduced star formation and the associated feedback from supernovae and showed that, given plausible assumptions, the cooling catastrophe can be avoided. The main deficiency in the semi-analytic program is that it cannot describe the geometry of mergers which is exceptionally important in the assembly of galaxies.

Smoothed Particle Hydrodynamic simulations of galaxy formation (Katz, 1992; Navarro and White, 1994; Evrard *et al.*, 1994, for example) detail the hierarchical merging history, but have been limited in terms of resolution. Achieving high resolution is particularly difficult. Since galaxy formation is affected by long range tidal forces, the simulation must be large enough to include these, which in turn enforces a low mass resolution. Two solutions to this problem exist; the first samples the long range fields using lower particle resolution than the main simulation region (the multiple mass technique, Porter, 1985) while the second includes the long range fields as a pre-calculated low resolution external field (*e.g.* Sommer-Larsen *et al.*, 1998*b*). Simulations performed in this manner represent lengths scales from 50 Mpc to 1 kpc, a dynamic range of 5×10^4.

It is well known that in SPH simulations it is comparatively easy to form flattened disc structures resembling disc galaxies. However, the resulting gaseous structures are deficient in angular momentum (see Navarro, Frenk and White, 1995, for a comparison of a number of simulations to observed galaxies). The loss of angular momentum occurs during the merger process as dense gas

cores lose angular momentum to the dark matter halos (see Barnes, 1992, for an explanation of the mechanism). This is a very significant problem since a fundamental requirement of the disc formation model, presented in Fall and Efstathiou (1980), is that the gas must maintain a similar specific angular momentum to the dark matter to form a disc. The solution to this problem is widely believed to be the inclusion of star formation and feedback from supernovae and stellar winds. By including these effects the gas should be kept in a more diffuse state which consequently does not suffer from the core-halo angular momentum transport problem. Notably, a recent letter by Dominguez-Tenreiro *et al.* (1998), based upon analytic work by Christoudolou *et al.* (1995) and van den Bosch (1998), has shown that the inclusion of star formation may go some way in helping to resolve the angular momentum problem. They suggest that a second mechanism, bar formation, also contributes to the loss of angular momentum within the disc.

Because of the freedom to chose how energy is distributed in the hydrodynamic sector of the simulation—thermal or kinetic are possible—it is exceptionally difficult to decide how the feedback energy should be distributed. As a direct result of this freedom, a number of different algorithms for the distribution of feedback energy have been proposed (Katz, 1992; Navarro and White, 1993; Mihos and Hernquist, 1994). Since the interstellar medium is multi-phase (McKee and Ostriker, 1977), the feedback process should be represented as an evaporation of molecular clouds. Note that in a single-phase model, the analogue of the cloud evaporation process is thermal heating. A seminal attempt at representing this process has been made by Yepes *et al.* (1997), and recently Hultman and Pharasyn (1999) have adapted the Yepes *et al.* model to SPH. At the moment it is unclear how well motivated these models are since (1) they must make a number of assumptions about the physics of the ISM and (2) they are not truly multiphase, since the dynamics are still treated using a single phase. Consequently, given the uncertainties inherent in multiphase modeling, the investigation presented in this chapter explores a single phase model.

As a first approximation, the star formation algorithms can be divided into three groups. The first group contains algorithms which rely upon experimental laws to derive the star formation rate (Mihos and Hernquist, 1994, for example). The second group contains those which predict the SFR from physical criteria (Katz, 1992, for example). The third group is comprised of algorithms which do not attempt to predict the SFR, but instead set a density criterion for the gas so that when this limit is reached the gas is converted into stars (Gerritsen and Icke, 1997, for example). In this chapter the first approach is taken.

Given that feedback occurs on sub-resolution scales, it is difficult to decide upon the scale over which energy should be returned. However, SPH incorporates a minimum scale automatically - the smoothing scale. Katz (1992) was the first to show that simply returning a specified amount of energy to the ISM is ineffective: radiative cooling at high density is too efficient. This effect is primarily because of the relatively high density of individual SPH particles in 'galaxies' in simulations ($n_H > 1$ cm^{-3}). It is further complicated by the fact that SPH cannot respond quickly to a rapid injection of energy, at least compared to the time-scale of star formation, which is of order one Myr. In this chapter a new thermal feedback model is presented in which the radiative losses are reduced by changing the density used in the radiative cooling equation. The density used is predicted from the ideal gas equation of state and then integrated forward, decaying back to the SPH density (in a prescribed period). The effect of preventing radiative losses using a brief adiabatic period of evolution for the feedback region (Gerritsen, 1997) is also examined. An alternative method of returning thermal energy is to heat an individual SPH particle (Gerritsen, 1997). This method has been shown to be effective in simulations of isolated dwarf galaxies. The final mechanism considered is one that attempts to increase the energy input from SN to account for the high radiative losses. Mechanical feedback boosts are not considered for the following reasons; (1) parameters that are physically motivated (*e.g.* feedback efficiency of 10%) appear to reproduce unphysical results, (2) the method makes no account for force softening, (3) Gerritsen (1997) has shown that these models produce excessive velocity dispersion in discs.

The response of the (simulated) ISM to a single feedback event is examined to gain an understanding of the qualitative and quantitative performance of each feedback algorithm. To determine whether the parameters of the star formation are more important than the dynamics of the simulation, an exploration of the parameter space of one of the algorithms is undertaken. Since feedback is expected to have a more significant effect on dwarf systems (due to the lower escape velocity) the effect of the feedback algorithms on a Milky Way prototype is contrasted against a model of the dwarf galaxy NGC 6503. Following this investigation, a series of cosmological simulations is conducted to examine whether the conclusions from the isolated simulations hold in a cosmological environment. Particular attention is paid to the rotation curves and angular momentum transport between the galaxy and halo.

The structure of the chapter is as follows. In section 3.2, important features of the numerical technique are reviewed. In sections 3.3-3.4, the star formation prescription is presented and a detailed analysis of its performance on isolated test objects presented conducted. Results from simulations are reviewed in section 3.4, and a summary of the findings is given in section 3.6.

3.2 Numerical Method

An explicit account of the numerical method, including the equation of motion and artificial viscosity used, is presented in chapter 2. The main points are summarized for clarity. Gravitational forces are evaluated using the adaptive Particle-Particle, Particle-Mesh algorithm (AP^3M,Couchman, 1991). Hydrodynamic evolution is calculated using SPH. Notable features of algorithm relevant to the hierarchical simulations follow,

- The neighbour smoothing is set to attempt to smooth over 52 neighbour particles, usually leading to a particle having between 30 and 80 neighbours.
- The minimum hydrodynamic resolution scale is set by $h_{min} = \epsilon/2$ where ϵ is the gravitational softening.
- Once a particle reaches h_{min}, *all* particles within $2h_{min}$ are smoothed over. Thus at this scale the code changes from an adaptive to nonadaptive scheme.

Radiative cooling is implemented using an assumed 5% $Z_\odot$ metallicity. The precise cooling table is interpolated from Sutherland and Dopita (1993). Radiative cooling is calculated in the same fashion as discussed in chapter 2.

3.3 Star Formation Prescription

3.3.1 Implementation details

The star formation algorithm is based on that presented in Mihos and Hernquist (1994). Kennicutt (1998) has presented a strong argument that the Schmidt Law, with star formation index $\alpha = 1.4 \pm 0.15$, is an excellent model. It characterizes star formation over *six decades of gas density.* Given this result it was decided not to employ a model that predicts the star formation rate on the basis of the Jeans' collapse model (Katz, 1992), the accuracy of which is unknown.

For computational efficiency, a Lagrangian version of the Schmidt Law is used, that corresponds to a star formation index $\alpha = 1.5$,

$$\frac{dM_*}{dt} = C_{sfr}\, \rho_g^{1/2}\, M_g, \tag{3.1}$$

where C_{sfr} is the star formation rate normalization, the g subscripts denotes gas and the $*$ subscript stars. Assuming approximately constant volume over a time-step, both sides may divided by the volume and the standard Schmidt Law with index $\alpha = 1.5$ is recovered. A value of $\alpha = 1.5$ is preferred since it leads to the square root of the gas density, which is numerically efficient and the value is within the error bounds. As written the constant C_{sfr} has units of inverse root density time, but can be made dimensionless by multiplying by $(4\pi G)^{1/2}$. Hence, equation 3.1 may be written with dimensionless constants, leading to the following form,

$$\frac{dM_*}{dt} = \sqrt{4\pi G}\, c_*\, \rho_g^{1/2}\, M_g, \tag{3.2}$$

where c_* is the dimensionless star formation rate (Katz, 1992). The range for this parameter is reviewed in section 3.3.4.

Star formation is allowed to proceed in regions that satisfy the following criteria,

1. the gas exceeds the density limit of 2×10^{-26}g cm^{-3}
2. the flow is convergent, ($\nabla . \mathbf{v} < 0$)
3. the gas temperature is less than 3×10^4 K
4. the gas is partially self-gravitating, $\rho_{gas} > 0.4\rho_{dm}$

The first criterion associates star formation with dense regions (regardless of the underlying dark matter structure). The second criterion is included to link star formation with regions that are collapsing. The third prevents star formation from occurring in regions where the average gas temperature is too high for star formation. The final criterion is particularly relevant to cosmological simulations since it limits star formation to regions where the dynamics are at least partially determined by the baryon density. This requirement is motivated by the trivial observation that, since star formation occurs in galaxies, it is tied to baryon dominated regions.

Representing the growth of the stellar component of the simulation requires compromises. It is clearly impossible to add particles with masses dM_* at each time-step since this would lead to many millions of small particles being added to the simulation. Alternatively, a gas particle may be viewed as having a fractional stellar mass component, *i.e.* the particle is a gas-star hybrid (in the terminology of Mihos and Hernquist, 1994). Thus, as gas is turned into stars, the stellar mass increases while the gas mass decreases. Mihos and Hernquist take this idea to its limit by only calculating gas forces using the gas mass of a particle (gravitational forces are unaffected because of mass conservation). A drawback of this method is that it couples the dynamics of the stars and gas, which is highly undesirable. The scheme used in this work is as follows: Once M_* has been evaluated, the associated mass increase over the time-step is added to the 'star mass' of the hybrid gas-star particle. Two star particles (of equal mass) can be created for every gas particle. This is to ensure that the feedback events occur frequently, thus making the process comparatively smooth, and at the same time prevents the spawning of too many star particles. The creation of a star particle occurs when the star mass of a gas-star particle reaches one half that of the mass of the *initial* gas particles. The gas-star particle mass is then decremented accordingly. The second star particle is created when the star mass reaches 80% of half the initial gas mass. SPH forces are calculated using the total mass of the hybrid particles. Note, in all the following sections, "gas particle" should be interpreted meaning "gas-star hybrid particle". These assumptions yield a star formation algorithm where each gas particle has a star formation history. This assumption is motivated because cloud complexes in any galaxy have an associated star formation history.

There are two notable drawbacks to this algorithm. Firstly, since star particles are only formed when the star mass exceeds a certain threshold there is a delay in forming in stars. As a consequence, prior to the star particle being formed, the SPH density used in the calculation of the SFR will be overestimated. By selecting the star mass to be one half that of the gas mass this problem is reduced, but it is not removed.

Once a star particle is created the associated feedback must be evaluated. A simple prescription is utilized, namely that for every $100M_\odot$ of stars formed there is one supernovae which contributes 10^{51} erg to the ISM. This value is used in Sommer-Larsen, Gelato and Vedel (1998*b*), and gives the following: 5×10^{15} erg g^{-1} of star particle is fed back to the surrounding ISM. Since this value is a specific energy, a temperature can be associated with feedback regions (see section 3.3.2). Navarro and White (1993), using a Salpeter IMF, with power law slope 1.5 and mass cut-off at 0.1 and 40 $M_\odot$, derive that 2×10^{15} erg g^{-1} of star particle created is fed back to the surrounding gas. The actual value is subject to the IMF, but scaling between values can be achieved using a single parameter. Only brief attention is paid to changing this parameter since it is more constrained than the others in the model. Variations in metallicity caused by the feedback process (Martel and Shapiro, 1998) are not considered, as this involves a complicated feedback loop involving the gas density and cooling rate.

3.3.2 Returning feedback energy

The following sections describe each of the feedback algorithms considered.

Energy smoothing (ES)

The first of the methods is comparatively standard: the total feedback energy is returned to the local gas particles. For simplicity in the following argument, it is assumed a tophat kernel is used to feedback the energy over the nearest neighbour particles. The number of neighbours is ensured to be between 30 to 80 which may (rarely) divorce the feedback scale from the minimum SPH smoothing scale set by h_{min}. Given these assumptions for star formation, and working in units of internal energy, it is possible to evaluate the temperature increase, ΔT of a feedback region as follows,

$$\Delta T = \frac{2}{3}\frac{\mu m_p}{k}\frac{E_{SN}M_*}{N_*M_g}, \tag{3.3}$$

which given $N_* = 52$, $E_{SN} = 5 \times 10^{15}$ erg g^{-1}, and $M_*/M_g = 1/2$ yields,

$$\Delta T \simeq 2.4 \times 10^7 \frac{M_*}{N_*M_g} \simeq 2.3 \times 10^5 \text{ K}. \tag{3.4}$$

Clearly this boost may be increased or decreased by altering any one of the variables E_{SN}, N_* and the ratio M_*/M_g. Keeping E_{SN} as is, but reducing N_* to 32 and increasing M_*/M_g to 1 would yield $\Delta T \simeq 7 \times 10^5$ K. This demonstrates the sensitivity of feedback to the SPH smoothing scale.

Single particle feedback (SP)

Alternatively, all of the energy may be returned to a single SPH particle. Gerritsen (1997) has shown that this is an effective prescription in simulations of evolved galaxies, yielding accurate morphology and physical parameters. In this case the temperature boost is trivially seen to be

$$\Delta T \simeq 2.4 \times 10^7 \mathrm{K}. \tag{3.5}$$

as $N_s = 1$ and $M_*/M_g = 1$. There is one minor problem in that when the gas supply is exhausted, the mechanism has no way of returning the energy (unless one continues to make star particles of smaller and smaller masses). As a compromise the nearest SPH particle is found and the energy is given to this particle.

Temperature smoothing (TS)

The final feedback mechanism considered is one that accounts for the fast radiative losses by increasing the energy input. The first step is to calculate the temperature a single particle would have if all the energy were returned to it (as in the SP model). Then this value is smoothed over the local particles using the SPH kernel. This method leads to a vastly higher energy input than the others and thus represents a useful model since it provides a means for examining what can happens when extreme feedback is applied. For the isolated simulations, the cooling mechanism (see below) was not adjusted in this model since the feedback regions in the disc have cooling time of more than a handful of time-steps (at least greater than 10). In the cosmological simulations, the alternative cooling mechanisms were considered since the cooling time is only a few time-steps (Katz, 1992; Sommer-Larsen *et al.*, 1998*b*).

Preventing immediate radiative energy losses

The first method for preventing the immediate radiative loss of the feedback energy is to alter the density value used in the radiative cooling mechanism. This change is motivated by Gerritsen's (1997) tests on turning off radiative cooling in regions undergoing feedback. Assuming pressure equilibrium between the ISM phases (which is not true after a SN shell explodes but is a good starting point) one may derive the estimated density that the region would have after the SN shell has exploded. If the local gas energy is increased by E_{SN}, then the perfect gas equation of state yields

$$\rho_{est} = \frac{E_i \rho_i}{E_i + E_{SN}}. \tag{3.6}$$

Following a feedback event the estimated density is allowed to decay back to it's local SPH value with a half-life $t_{1/2}$. To calculate the decay rate (*i.e.* to predict the cooling density at the time-step $n+1$ from that at time-step n) the following function is used

$$\rho_{cool}^{n+1} = \begin{cases} \rho_{cool}^n + dt \times \Delta\rho^n (t_f^2/t_{1/2}^3) e^{-0.33(t_f/t_{1/2})^3}, & t_f \leq t_{1/2}; \\ \rho_{SPH}^n - \Delta\rho^n e^{-0.693 dt/t_{1/2}}, & 3t_{1/2} \leq t_f < t_{1/2}; \\ \rho_{SPH}^{n+1}, & t_f > 3t_{1/2}\,, \end{cases} \tag{3.7}$$

where t_f is time since the feedback event occurred, $\Delta\rho^n = \rho_{SPH}^n - \rho_{cool}^n$, and dt is the time-step increment. Once a region passes beyond $3t_{1/2}$, cooling density is forced to be equivalent to the SPH density, although usually the values converge within $2t_{1/2}$. This comparatively complex function is chosen because in a simple exponential decay model, the cooling density increases by the largest amount immediately following the feedback event. To have any effect the cooling density must be allowed to persist at its low value for a reasonable period of time. In Figure 3.1 the two densities calculated after a single feedback event are compared. This cooling mechanism is denoted by an *na* suffix (for non-adiabatic) on the energy input acronym. Springel and White (in prep) have considered a similar model (Pearce, private communication).

Gerritsen allowed his SPH particles to remain adiabatic for 3×10^7 years, approximately the lifetime of a $8M_\odot$ star. It is difficult to argue what this value should be and hence the $t_{1/2}$ parameter space was explored. Note that if during the decay period another feedback event occurs in the local region, the density value used is the minimum of the current decaying one and the new calculated value.

As a second method for preventing radiative losses Gerritsen's idea was utilized: make the feedback region adiabatic. This is achieved by using the same mechanism that calculates the estimated

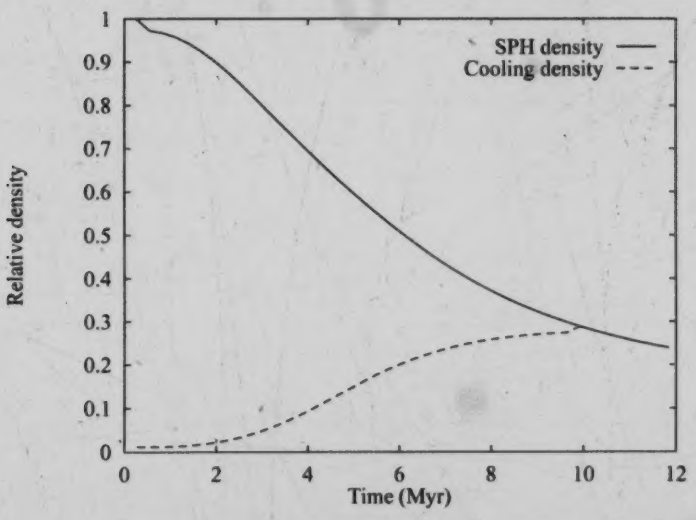

Figure 3.1: Evolution of the cooling density and the SPH density following a single feedback event in the ESna scheme. $t_{1/2} = 5$ Myr, and clearly the values converge within $2t_{1/2}$. Note the small initial drop in the SPH density in response to the feedback energy, followed by a slower expansion. The initial cooling density is approximately 5% of the SPH density, consistent with the feedback temperature of 2×10^5 K in an ambient 10,000 K region.

density value. Provided this value is less than half that of the local SPH value then the particle is treated as adiabatic. Above this value the estimated density is used. This cooling mechanism is denoted with an *a* suffix (for adiabatic) on the energy input acronym.

3.3.3 Comparison of methods

To test each of the feedback methods and gain insight into their affect on the local ISM, a single feedback event was set up within a prototype isolated Milky Way galaxy. The evolution of the particles within $3h$ of the feedback event were followed. The time evolution of the temperature versus radius for each scheme is shown in Figure 3.2.

Qualitative discussion

The adjusted cooling mechanism has little effect on the ES run because the estimated density is not low enough to increase the cooling time significantly beyond the length of the time-step. Including a prefactor of 0.1 in the equation, so that the estimated density is significantly lower, does increase the cooling time sufficiently. However, since introducing the adiabatic phase allows the feedback energy to induce expansion, we prefer this method, rather than trying to adjust the estimated density method. For the SP run the density reduction is much higher (since the total energy, E_{SN}, is applied to the single particle) and hence the mechanism does allow the particle to remain hot. There is little perceptible difference between the SPa and SPna profiles.

Both SP feedback, TS, and ESa induce noticeable expansion of the feedback region. Thus after the heat input has been radiated away, the continued expansion introduces adiabatic cooling (since the temperature of the region falls below the 10,000 K cut off of the cooling curve). It is particular noticeable in the TS plot where the low temperature plateau continues to widen quite drastically and this is manifest in the simulation with the appearance of a large bubble in the disc. Caution should be exercised in interpreting these bubbles in any physical manner, since the size of them is set solely by the resolution scale of the SPH. ESa also produces bubbles, but due to the lower temperature there is less expansion. Single particle feedback produces the smallest bubbles and often ejects the hot particle vertically from the disc. This occurs with regularity since the smoothing scale is typically larger than the disc thickness: if the hot particle resides close to the edge of the disc the pressure forces from the surrounding particles will be asymmetric resulting in a strong 'push' out of

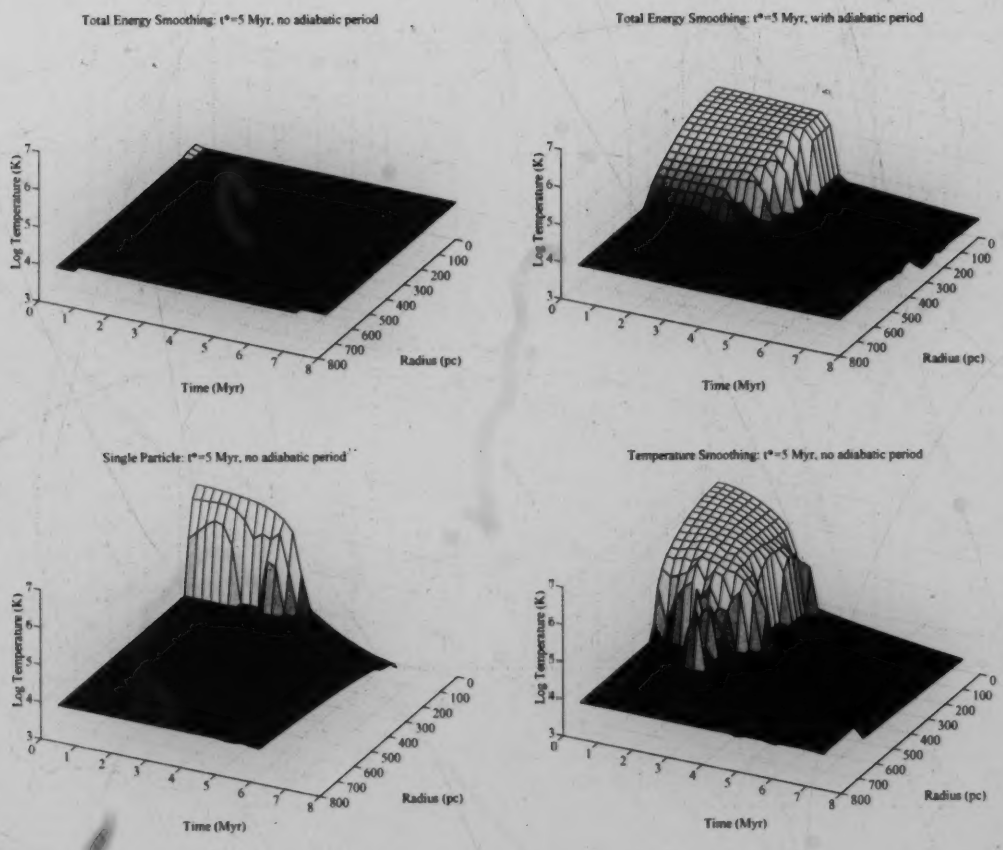
Total Energy Smoothing: t*=5 Myr, no adiabatic period
Total Energy Smoothing: t*=5 Myr, with adiabatic period
Single Particle: t*=5 Myr, no adiabatic period
Temperature Smoothing: t*=5 Myr, no adiabatic period
Log Temperature (K)
Time (Myr)
Radius (pc)

Figure 3.2: Comparison of different feedback schemes. The temperatures of particles within $3h$ of the feedback particle have been interpolated on to a grid. The total energy return normalization, e^*, was set to 0.4, corresponding to 2×10^{15} erg g^{-1} of stars created, although the qualitative conclusions for $e^* = 1.$ remain the same. There are a number of noticeable differences between schemes. Firstly, even with the adjusted cooling density, the ES (energy smoothing) algorithm leads to very fast dissipation of the heat input and there is little effect on the surrounding gas. Including the adiabatic period allows the region time to expand, and at the end of the feedback period a small amount of expansion continues which is the cause of the temperature drop. The SP (single particle) feedback method (note the adiabatic version and non-adiabatic versions are almost identical) also leads to adiabatic cooling after the feedback energy has been radiated away, although the effect is localized around the feedback particle. TS (temperature smoothing) produces very rapid expansion and the region continues to undergo strong expansion after the feedback event.

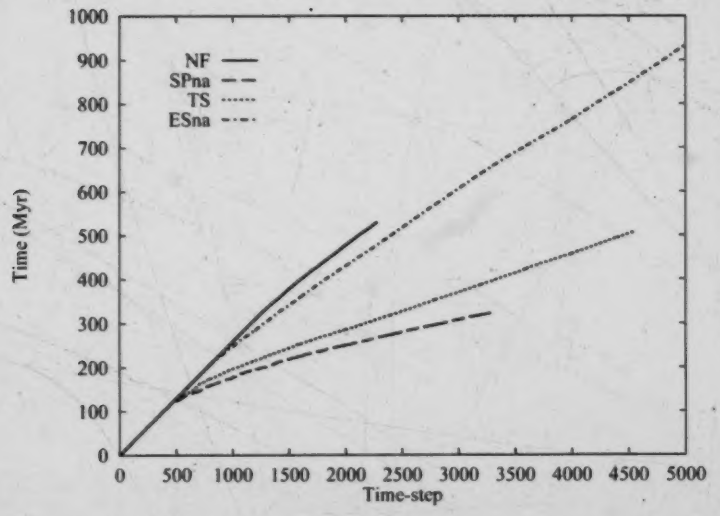

Figure 3.3: Effect of feedback scheme on the time-step selection in the simulation. Data from the Milky Way prototype runs is shown. Twice as many time-steps are required for SP feedback compared to runs without feedback. The ESa and SPa runs are not shown, but are approximately 5% lower than the respective runs without the adiabatic period.

the disc. Note that this is phenomenologically similar to the mechanism by which SN gas is ejected from discs (McKee and Ostriker, 1977), although this should not be over-interpreted.

The only scheme which stands out in this investigation is ESna: the cooling mechanism fails to prevent drastic radiative losses. The remainder of the algorithms produce an effect on both the thermal properties of the ISM and its physical distribution.

Effect of methods on time-step criterion

Since the code does not have multiple time-steps, it is important to discern whether one method allows longer time-steps than another. This is a desirable feature since it reduces the wall-clock time for simulations. Of course an algorithm which has a fast wall-clock time but produces poor results would never be chosen, however for two algorithms with similar results this criterion provides a useful parameter to choose one over the other. In Figure 3.3 a comparison of SPna, TS, ESna and the no feedback (NF) run is displayed.

The simulation time versus the number of time-steps was compared for data from the Milky Way prototype runs (see section 3.4.1). The SP methods produce the shortest time-step, requiring almost double the number of time-steps than the NF model. The ES variants require only 10% more time-steps than the run without feedback. In SP feedback the acceleration felt by the hot particle limits the time-step significantly. TS also requires more time-steps than ES due to the rapid expansion of feedback regions. Typically though, the number of time-steps required is somewhat less (20%) than that for single particle feedback.

3.3.4 Parameter space of the algorithm

All models exhibit dependencies on the free parameters c^* and e^*, corresponding to the SFR normalization and efficiency of the feedback energy return. The models which use a modified cooling formalism also exhibit a dependence upon $t_{1/2}$, the approximate half-life of feedback regions. To simplify matters, an ensemble with $e^* = 0.4$ (the value used in Navarro and White 1993) was run, and hence the effect of the c^* and $t_{1/2}$ could be concentrated upon. Then to determine the effect of e^*, two more simulations with $e^* = 1$ were run.

Run	c^*	$t_{1/2}^a$	e^*	N_{step}^b
5001	0.001	0	0.4	1999
5002	0.003	0	0.4	2001
5003	0.01	0	0.4	1924
5004	0.03	0	0.4	1977
5005	0.1	0	0.4	1989
5006	0.3	0	0.4	2140
5007	1.0	0	0.4	2087
7001	0.033	1.	0.4	1989
7002	0.033	5.	0.4	2000
7003	0.033	10.	1.0	1947
7004	0.033	1.	1.0	1938
7005	0.033	10.	0.4	1940

Table 3.1: Summary of star formation parameter space simulations. The simulations were of a rotating cloud collapse (see chapter 2). [a] $t_{1/2} = 0$ denotes that feedback was removed from the simulation. [b] The number of steps are given to t=1.13, the final point of the parameter space plot.

Simple collapse test

To gain an understanding of the algorithms in a simple collapse model (that also may be run in a short wall-clock time) the rotating cloud collapse model of Navarro and White (1993, also see chapter 2) was utilized. Such models actually bear little resemblance to the hierarchical formation picture, but they do allow a fast exploration of the parameter space.

For this test the self-gravity requirement was removed. The reason for this is that in cosmological simulations it is virtually guaranteed that the gas in a compact disc will be self-gravitating. This is due to the low number of dark matter particles in the core of the halo relative to the number of gas particles.

The most important parameter in the star formation model is the c^* parameter since it governs the SFR normalization. Therefore, an ensemble of models was ran with $c^* \in [0.001, 1]$ (and $e^* = 0.4$). The secondary parameter in the model, $t_{1/2}$, is expected to have comparatively little effect on the star formation rate (due to the low volume factor of regions undergoing feedback). Hence only a range of plausible alternatives were considered, namely $t_{1/2} = 1, 5, 10$ Myr, in the ESa model. As a further test, e^* was increased to 1 and two additional simulations were ran with $t_{1/2} = 1$ and 10 Myr. The motivation behind this being to see whether e^* was too small to have a significant effect on the SFR regardless of the cooling mechanism used. The simulation parameters are detailed in table 3.1.

Results

Figure 3.4 displays a plot of the c^* parameter space, showing SFR and c^* versus time. Feedback was effectively turned off in this simulation by reducing the energy return efficiency to 10^{-7}. Although a severe amount of smoothing has had to be applied (a running average over 40 time-steps, followed by linear interpolation on to the grid) there are a number of interesting results.

The time at which the peak SFR occurs is almost constant across all values of c^*. This is an encouraging result - it indicates that the time at which star formation peaks is dictated by dynamics and not by the parameters of the model (at least without feedback). In fact the SFR peak time corresponds to the time when the collapse model reaches its highest density, following this moment a significant amount of relaxation occurs and the gas has a lower average density. This result is shown graphically in Figure 2.9 of chapter 2. In view of the lack of feedback and the idealized nature of the collapse, this result should be treated with caution.

Figure 3.5 shows the dependence of the SFR on the $t_{1/2}$ and e^* parameters. To detect trends in the SFR, a running average is shown, calculated over 40 time-steps. Clearly, there is little distinction between the runs with $t_{1/2} = 1$ and 10. This can be attributed to the low volume fraction of regions undergoing feedback. It is interesting to note that the SFR is more sensitive to the amount of energy returned, dictated by the e^* parameter, than the lifetime for which this energy is allowed to persist. The line corresponding to $e^* = 1$ (the standard energy return value of 5×10^{15} erg g^{-1}) does not have the secondary and tertiary peaks in the SFR exhibited by the $e^* = 0.4$ runs. This is probably attributable to the $\nabla.\mathbf{v}$ criterion: the $e^* = 1$ run produces enough heating to provide a significant

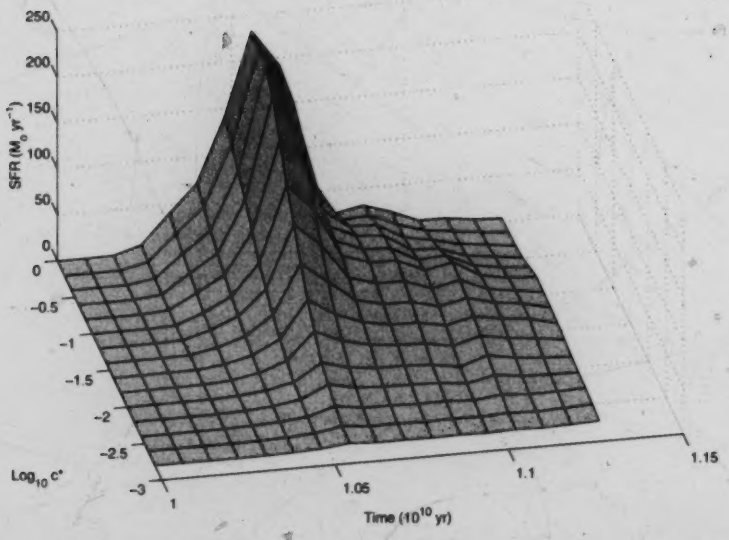

Figure 3.4: Dependence of the SFR on the c^* parameter in a model with no feedback. The data for seven runs was linearly interpolated to form the plotted surface. The time of the peak SFR moves very little with changing c^*, and almost all the models can be fitted to exponential decay models following the peak SFR epoch.

amount of expansion rendering $\nabla.\mathbf{v} \gg 0$ for the first feedback region. Note that since less of the gas is used at early times, the SFR at later epochs is higher.

In summary, while the c^* parameter clearly sets the SFR normalization, it does not change the epoch of peak star formation. Further, the $t_{1/2}$ parameter has little effect on the overall SFR due to the low volume fraction of feedback regions in the evolved system.

3.4 Application to isolated 'realistic' models

This section reports the results of applying the algorithm to idealized models of mature isolated galaxies. These models are created to fit the observed parameters of the system, *i.e.* the rotation curve and disc scale length. The characteristics of each model are discussed within the sections devoted to them. In this investigation the star formation epoch is quiescent and disc based, and thus not as rapid as that which would be expected in the early stages of the cosmological simulations (section 3.5). Both models were supplied by Dr. Fabio Governatto. A summary of the simulations is presented in table 3.2.

3.4.1 Milky Way prototype

The first prototype model is an idealized one of the Galaxy. It is desired that the model should reproduce the measured SFR ~ 1 $M_\odot$ yr^{-1} and also the velocity dispersion in the disc. Evolved galaxies have a lower relative gas content than protogalaxies. Further, because of hierarchical clustering, they are significantly more massive. Hence feedback is expected to have less effect on this model than on protogalaxies formed in simulations.

Model Parameters

The Milky Way prototype contains stars, gas and dark matter. The total mass of each sector is 5×10^{10} $M_\odot$, 9×10^9 $M_\odot$, and 3×10^{11} $M_\odot$ respectively. 11980 particles were used to represent the stars, 10240 to represent the gas and 10240 to represent the stars. The individual particle masses

Run	Simulation object[a]	feedback[b]	N^{c}_{step}	N_{SPH}
1001	NGC 6503	none	3010	10240
1002	NGC 6503	SPna	3453	10240
1003	NGC 6503	SPa	3986	10240
1004	NGC 6503	TS	5345	10240
1005	NGC 6503	ESna	3453	10240
1006	NGC 6503	ESa	4535	10240
2001	Milky Way	none	387	10240
2002	Milky Way	SPna	952	10240
2003	Milky Way	SPa	943	10240
2004	Milky Way	TS	723	10240
2005	Milky Way	ESna	424	10240
2006	Milky Way	ESa	453	10240
6001	Cosmological	SPa	4792	17165
6002	Cosmological	SPna	4710	17165
6003	Cosmological	ESa	4319	17165
6004	Cosmological	ESna	4319	17165
6005	Cosmological	TSna	4335	17165
6006	Cosmological	TSa	4322	17165
6007	Cosmological	NF	4341	17165
6008	Cosmological	NSF	4314	17165
6010	Cosmological	TS	4342	17165

Table 3.2: Summary of the main simulations using realistic models. [a]'Cosmological' refers to the object formed being derived from a cosmological simulation. [b]SPa=Single particle adiabatic period, SPna=single particle no adiabatic period but adjusted cooling density, ESna=Total energy smoothing with adjusted cooling density but no adiabatic period, ESa=Total energy smoothing with adiabatic period, TS=Temperature smoothing (normal cooling), NF=no feedback, NSF=no star formation. [c]For the cosmological simulations the number of time-steps to $z = 1.0$ is given. Since the isolated simulations were not run to the same final time, the average number of time-steps per 100 Myr is shown.

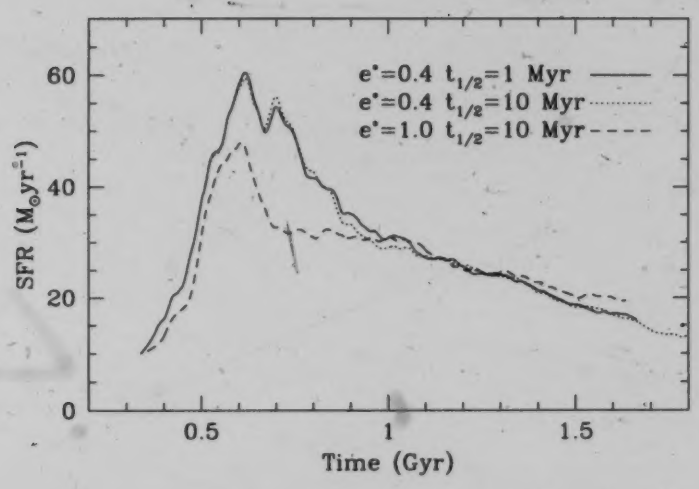

Figure 3.5: Dependence of the SFR on the $t_{1/2}$ and e^* parameters (for the ESa algorithm). Comparing the runs with the same e^* parameter shows that varying $t_{1/2}$ from 1 to 10 has no noticeable effect. Conversely, changing e^* from 0.4 to 1.0 removes the secondary and tertiary peaks in the SFR.

were 4×10^6 $M_\odot$, 9×10^5 $M_\odot$, and 3×10^7 $M_\odot$ respectively. The (stellar) radial scale length was 3.5 kpc and scale height 0.6 kpc. Density and velocities were assigned using the method described in Hernquist (1993). The maximal radius of the dark matter halo was 85 kpc. The artificial viscosity was not shear-corrected in this simulation.

A comparatively large softening length of 0.5 kpc was used, rendering the vertical structure of the disc poorly resolved. However, this is in-line with the softening lengths that are typically used in cosmological simulations (of order 2 kpc). Shorter softening lengths allow higher densities in the SPH, which in turn leads to higher SFRs. The self-gravity requirement was again removed.

Results

Due to the finite size of the computational grid, the code was unable to follow all of the simulations to the desired final epoch (500 Myr). This limitation was most noticeable in the SP simulations where ejected gas particles reached the edge of the computational domain within 320 Myr. It is possible to remove these particles from the simulation since they are comparatively unimportant to the remainder of the simulation. However maintaining an accurate integration was considered to be more important.

In Figure 3.6, the gas particle distributions are shown for the no feedback, SPa, ESa and TS runs. Of the versions not shown, ESna has a smooth disc since the feedback regions do not persist as long and the SPna disc resembles that from the SPa run (see section 3.3.3). TS produces the most significant disturbance in the disc, which is to be expected given that it injects more energy into the ISM than the other methods. For TS feedback, 7% of the disc gas had been ejected (falls outside an arbitrary horizontal 6 kpc band) by t=323 Myr rising to 14% by t=506 Myr. Note that the amounted of ejected gas is measured relative to the total gas in the simulation at the time measurement (which is a decreasing function over time). Particle ejection velocities, v_z, were close to 300 km s^{-1}, although some did achieve escape velocity ($\simeq$ 500 km s^{-1} at solar radius). Hence, while TS can project particles out of the halo ('blow-away') it preferentially leaves them bound in the halo ('blow-out'). SP feedback (both SPa and SPna) has a similar evolution, but only tends to eject the single heated particle during each feedback event, thus leading to a lower mass loss rate (1% of the disc gas had been ejected by t=323 Myr for both versions). Particles were often ejected with $v_z > 600$ km s^{-1}, which is larger than the escape velocity, leading to a stronger relative tendency for blow-away. ESa also ejects particles from the disc (0.4% ejected by t=323 Myr), although in general the particles have lower velocities ($v_z \simeq 200$ km s^{-1}) than the particles ejected by either TS or SP. Hence, almost all the ejected gas remains bound to the system, *i.e.* ES leads only to blow-out. ESna does not eject particles since the feedback regions cool sufficiently fast (0.01% ejected by t=323

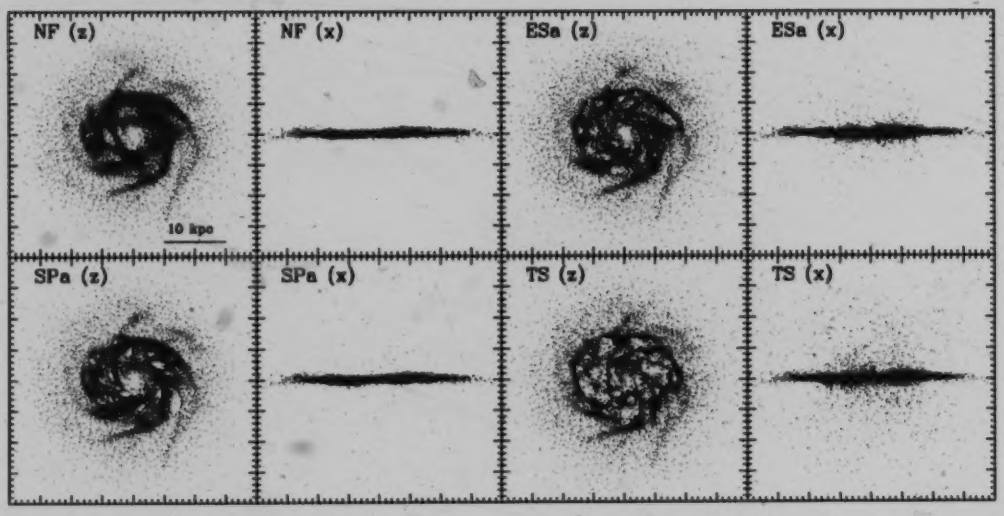

Figure 3.6: Morphology of Milky Way simulations at t=323 Myr. z- and x-projections are shown to detail disc and gas halo structure. Both SPa feedback and TS lead to significant ejection of matter from the disc. ESa 'inflates' the disc but does not eject as much matter as TS.

Myr).

The SFRs for three of the simulations (NF,TS and ESa) are plotted in Figure 3.7. Most noticeable is the reduction in the SFR produced by TS and ESa (TS is 35% lower than no feedback at t=500 Myr, ESa is 10% lower). This is due to three factors; (a) the ejection of matter from the disc depletes the cold gas available for star formation (see Figure 3.6), (b) smoothing feedback energy leads to spatially extended 'puffy' feedback regions in the disc, which in turn leads to a lower average density, and hence lower SFR, (c) particles in the feedback regions will typically be above the temperature threshold, which prevents star formation which further reduces the SFR. For single particle feedback the lower mass loss rate leads to a higher SFR than for the TS or ESa runs. Of the versions not plotted, ESna resembles the no feedback run (SFR approximately 3-4% lower on average), since most of the energy is rapidly radiated away. SPna resembles the SPa since the feedback events produce very similar effects (see section 3.3.3).

To calculate radial profiles of the discs, an arbitrary plane of thickness 6 kpc was centered on the disc. This thickness ensured that the stellar bulge was contained within the band. Radial binning was then performed on this data set using a cylindrical binning procedure.

In Figure 3.8, gas rotation curves are compared for the simulations at t=323 Myr. The rotation curve was calculated by radial averaging $|\mathbf{r} \times \mathbf{v}|/|\mathbf{r}|$ rather than by calculating the circular velocity from $\sqrt{GM(<R)/R}$. This method provides a fairly accurate depiction of the rotation curve that would be measured, although in the core regions where there are few particles in the bins, the measurement can become 'noisy'. Clearly, Figure 3.8 shows that there is little difference between schemes (a maximum of 9% at 4 scale lengths—ignoring the under-sampled central values). At large radii the curves match precisely since there are no feedback events in the low-density outer regions of the disc, except for the TS run where a feedback event has ejected particles to the outer regions. Comparing to the initial rotation curve (not shown), the disc has clearly relaxed, extending both in the tail and toward the center. The curves were also examined at t=507 Myr (for those simulations that could integrate that long) and similar findings were found with maximum differences being in the 10% range.

A more telling characteristic is the gas radial velocity dispersion. Unfortunately it is difficult to relate the measurements made here to those of molecular clouds (Malhotra, 1995), primarily because the mass scales are significantly different. Nonetheless, it is interesting to compare each of the separate feedback schemes. In Figure 3.8 the radial velocity dispersion, σ_r, is plotted for three of the simulations at t=323 Myr (again only considering matter within the 6 kpc band). Temperature smoothing produces a large amount of dispersion due to the excessive energy input (interior to 8 scale lengths it varies between being 40% to 300% higher than other values). Note that there is a

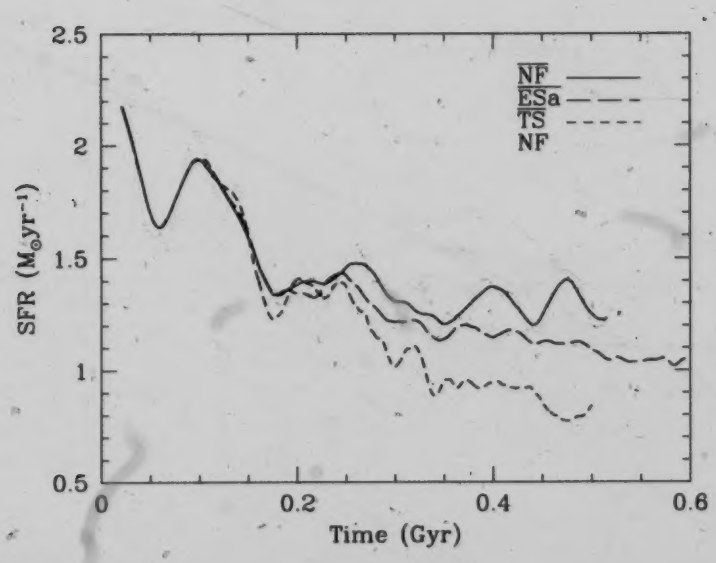

Figure 3.7: SFRs for the Milky Way prototype (time averaged over 160 time-steps to show trend). The SP variants are not shown since their evolution is similar to that of the ES version plotted. TS produces a significant (35% at t=500 Myr) reduction in the SFR as compared to no feedback. ES also reduces the SFR, but has a less significant effect (10% reduction at t=500 Myr versus no feedback) than TS.

direct correlation between a higher velocity dispersion and a lowered measured rotation curve. This is asymmetric drift: feedback events produce velocity dispersion which in turn increases the relative drift speed, v_a, between the gas and the local circular velocity (Binney and Tremaine, 1987). The remaining algorithms (SPa, SPna, ESa, ESna) exhibit similar velocity dispersions. Thus, for these variants, the bulk dynamics remain the most important factor in determining the velocity dispersion. Although caution has been advised in comparing the velocity dispersion presented here with that of the value for local molecular clouds, it is worth while noting that measurements for the Milky Way suggest $\sigma_{v_{cloud}} = 9 \pm 1$ km s^{-1}(Malhotra, 1995).

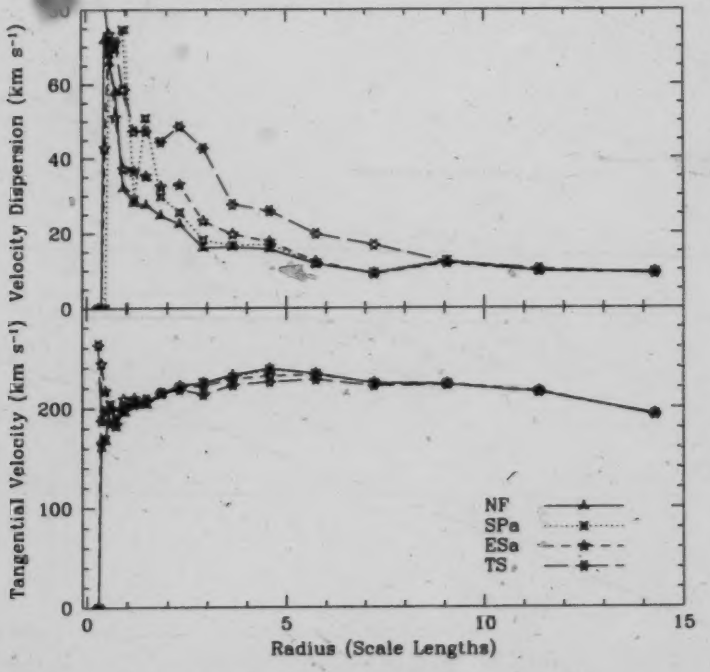

Figure 3.8: Rotation curves and radial velocity dispersions for the Milky Way prototype at t=323 Myr, for the NF, SPa, ESa and TS runs. There is only a marginal difference between rotation curves because the radial averaging smooths out the effect of inhomogeneous feedback regions. The TS gas exhibits a 8% reduction in the rotation curve at a radius of 4.5 scale lengths due to asymmetric drift. The ESa rotation curve is reduced by only 4% at this radius. The TS algorithm exhibits significantly higher velocity dispersion than any of the other variants. This is attributable to the large energy input driving winds that strongly affect the disc structure (visible in Figure 3.6 as large holes in the disc). All the other algorithms differ only very marginally.

Summary

Temperature smoothing (TS) is clearly the most violent feedback method, producing an SFR lower than the other algorithms and also tending to evaporate the disc. Given the comparatively large escape velocity of the Milky Way (lower limit of 500 km s^{-1} at solar radius), this is an unrealistic picture because such behavior is expected to be shown by dwarf systems. Of the remaining algorithms, the single particle (SP) versions produce reasonable physical characteristics, but are handicapped in terms of utility by a large number of time-steps. The energy smoothing variant with an adiabatic period (ESa) appears to be the best compromise in these simulations. It does not require an excessive number of time-steps while the disc morphology and evolution are within reasonable bounds: there is no excessive blow-out or blow-away.

3.4.2 Dwarf prototype

The second model is an idealized version of NGC 6503. Due to their low mass, and consequently lower escape velocity, dwarf systems are expected to be more sensitive to feedback. In simulations, the over-cooling problem suggests that to form a large disc system from the merger of small dwarfs, the dwarf systems must be significantly extended in space (ideally similar to that for an adiabatic system, Weil *et al.*, 1998). Feedback is currently believed to be the best method for achieving this. A lower bound on the escape velocity of the system is given by, $\sqrt{2}v_c$, (Binney and Tremaine, 1987). Hence, given $v_c \simeq 110$ km s^{-1}, the escape velocity must exceed 155 km s^{-1}. Note, though, that this value is less than half that for the Milky Way prototype, thus feedback should have a more significant effect on morphology.

Detailed numerical studies of NGC 6503 have been conducted by Bottema and Gerritsen (1997) and Gerritsen (1997). The motivation in this investigation is different to the previous ones which attempted to explain the observed dynamics of NGC 6503. In contrast, this investigation attempts to determine bulk properties at comparatively low resolution, in accordance with that found in cosmological simulations.

Model Parameters

As for the Milky Way prototype, this model contains stars, gas and dark matter. The total mass of each sector is 3×10^9 M$_\odot$, 1×10^9 M$_\odot$, and 5×10^{10} M$_\odot$ respectively. 10240 particles were used in each sector, yielding individual particle masses of 3×10^5 M$_\odot$, 1×10^5 M$_\odot$, and 5×10^6 M$_\odot$, respectively. The radial scale length of the simulation was 1.16 kpc, and the scale height 0.1 kpc. Gas density and particle velocities were assigned in the same fashion as the Milky Way prototype. The artificial viscosity was not shear-corrected.

Six simulations were run, each using a different method of feedback (including no feedback as the control experiment). The method used in each simulation and the number of time-steps per 100 Myr are summarized in table 3.1.

Results

As in the Milky Way simulations, the SP feedback models produced significant blow-away at the outset of the simulation and particles ejected from the disc rapidly escaped the halo. Consequently, the evolution of these systems had to be halted at very early times (close to 200 Myr). Of the remaining algorithms, only TS was not integrated to at least 500 Myr. Thus, the following analysis concentrates on the ES variants.

Figure 3.9 shows the distribution of gas particles in the x- and z-projections and also the z-distribution measured vertically in bins, at t=230 Myr (note that although sufficient time has elapsed for feedback events to occur the discs still exhibit virtually identical rotation curves). Due to the comparatively long softening (300 kpc) length used in the simulation, there is a significant amount of relaxation from the initially 'thin' distribution, which is approximately 250 pc wide. Since in the simulation code the SPH resolution is at least twice the gravitational softening length, the disc was expected to fatten to 600 pc. Figure 3.9 shows that this is observed (in the run with no feedback). Once feedback is included, and matter is ejected from the disc, the z-distributions exhibit strong kurtosis due to particles orbiting high in the potential well. The most severe example of this being the TS variant, which rapidly ejects particles leading to significant mass loss from the disc (both blow-away and blow-out occur). The TS algorithm has produced extremely large bubbles in the disc. One bubble had a radius of almost 0.7 kpc, which is 60% of the disc scale length, while for the Milky Way prototype larger bubbles had a radius of about 0.8 kpc, approximately 40% of the scale length. Caution is again advised against over-interpretation of the absolute size of these bubbles relative to the disc (since this is set by the SPH smoothing scale). However, the comparison of the dwarf versus the Milky Way model is valid since the particle resolution is approximately the same for both models. A comparison of the gas distributions for the ESa runs (dwarf *vs.* Milky Way) at 500 Myr shows that while in the dwarf system the gas density puffs up beyond the stellar component, it does not do this for the Milky Way prototype. As in the Milky Way runs, ES preferentially leads to blow-out, although by 500 Myr some particles were close to escaping the halo. These results show feedback has a more significant effect on the dwarf system.

To calculate radial profiles of the discs a plane of thickness 2 kpc was used, centered on the disc. As for the Milky Way prototype, the thickness was chosen to ensure that the stellar content was included within the band. Radial binning was then performed on this data set using a cylindrical binning procedure. At t=580 Myr the gas rotation curves for ESna, ESa and no feedback remain very similar (Figure 3.10). Both of the curves exhibit asymmetric drift relative to the run with no feedback. The maximum difference (external to a radius of one scale length) occurs at 1.6 scale lengths where the adiabatic variant has a rotation curve that is 10% lower than the no feedback run.

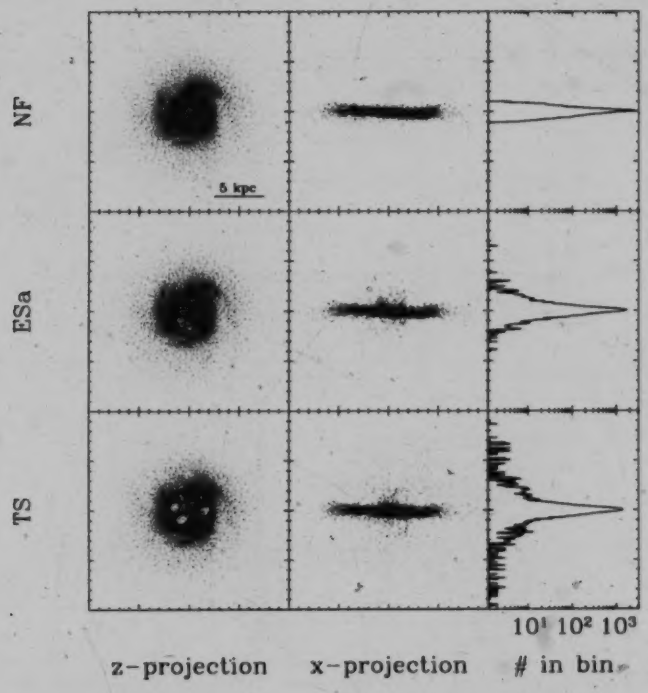

Figure 3.9: z-coordinate of gas particles in the NGC 6503 simulation binned into bins of width 0.150 kpc ($h_{min}/2$). A significant amount of relaxation is visible in the no feedback run, with the disc fattening to a width of $\simeq 600pc$. Feedback in the TS and ESa runs ejects particles into the halo and induces kurtosis in the z-distribution of particles.

At this radius the non-adiabatic run is only 3% lower than the run with no feedback. The outer edges of the distributions remain identical due to there being no feedback events in the low-density gas.

The gas radial velocity dispersion plot (Figure 3.10) shows that for this system ESa clearly introduces more dispersion (30% higher at a radius of 1.6 scale lengths). This is as expected: the combination of a lower escape velocity and comparatively long persistence of the feedback regions in the adiabatic variant allow the gas to escape to higher regions of the potential well. Additionally, bubble expansion in the plane of the disc persists for longer in the adiabatic variant. Notably, the run with no feedback shows an increasing velocity dispersion with radius. This can be attributed to the large softening length used, which in turn causes the SPH to smooth over a very large number of particles in the central regions (in excess of $10N_{smooth}$). Thus, in this region the gas distribution is dynamically cold.

Figure 3.11 shows the time-averaged SFRs for the three runs. By t=580 Myr the non-adiabatic run had ejected 1% of its mass from the disc (relative to the remaining gas), whilst the adiabatic run had ejected 2%. Examining the raw data shows that the strongest bursting is actually found in the no feedback run. The feedback in the other two runs keeps the disc more stable against local collapse. Of the data not plotted TS was similar to the ESa run, and had an SFR approximately 5% lower. By t=240 Myr 6% of the disc had been evaporated. Neither of the SP runs was integrated far enough to detect significant trends.

Summary

Although it was not possible to integrate all algorithms to the desired final epoch, it was still demonstrated that feedback does have a more significant impact on the dwarf system. This is evident both in the morphology (larger relative bubbles as compared the Milky Way prototype)

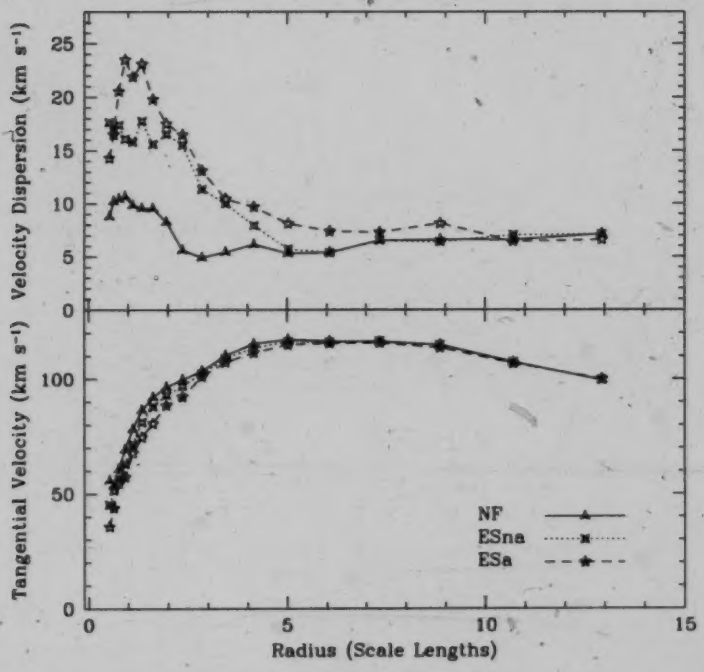

Figure 3.10: Comparison of rotation curves and radial velocity dispersions for the NGC 6503 dwarf simulation at t=580 Myr. There is little difference among all rotation curves, except in the nuclear region where feedback is more prevalent. Asymmetric drift is visible in the gas, with the ESa run having a lower rotation curve due to its higher velocity dispersion.

and radial characteristics (the radial velocity dispersion is far higher relative to the no feedback run). This increased sensitivity also allows differentiation between the adiabatic and non-adiabatic methods, which was at times difficult in the Milky Way prototype. Whether these conclusions can carry over to cosmological simulation is addressed in the next section.

3.5 Cosmological simulation

As discussed in the introduction, there are a number of problems that plague cosmological simulations of galaxy formation. This section examines the conjecture that feedback should be able to (1) prevent the overcooling catastrophe, thus suppressing early star formation and (2) prevent the angular momentum catastrophe, thereby allowing the formation of discs with specific angular momenta in agreement with observations. We study all of the feedback algorithms analysed in the previous sections and also include two new versions derived from combinations of the previously studied algorithms.

3.5.1 Initial conditions

Long-range tidal forces must be included in simulations because they have a significant effect on the evolution of galaxies. Unfortunately, a fixed resolution periodic box with equal number of dark matter and gas particles would require a prohibitively large number of particles. Hence the multiple mass technique (Porter, 1985) is used to overcome this problem.

In the SPH method, shocks are captured using an artificial viscosity. The artificial viscosity is turned on or off by the value of the **r.v** product between each pair of particles. The angular part of this product takes a maximal value if **r** and **v** are aligned, as is the case in collapse along a Cartesian

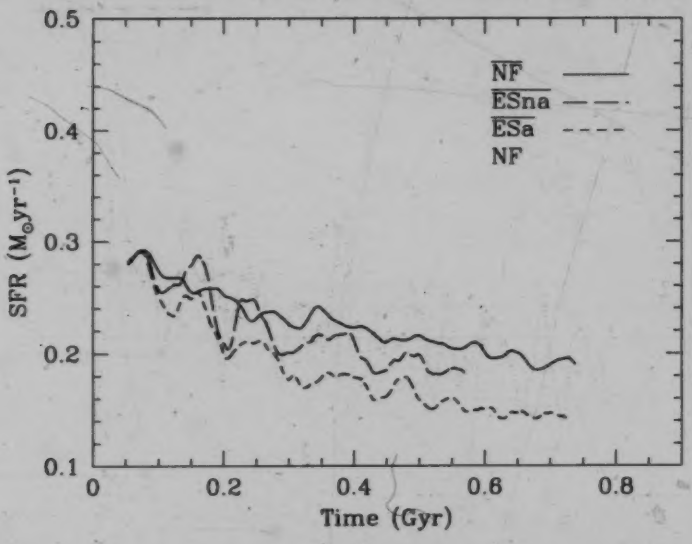

Figure 3.11: SFRs for the dwarf prototype. The ES runs and no feedback runs are shown since the remainder were not integrated for a sufficient time for conclusions to be drawn. The bar indicates that a running average over 320 time-steps was used to elucidate the trend in the SFR. Clearly the adiabatic run produces the lowest SFR, being approximately 20% lower than the non-adiabatic run.

grid. Collapse along a direction not aligned along the grid leads to scatter in the **r.v** dot product, and hence less shocking. Consequently, a set of initial conditions must be used that contains no preferred direction, which are termed 'glass-like'.

Given a hierarchical clustering scenario the first objects to form will have hundreds (at most) particles in them. Hence it is only necessary to create initial conditions which have no preferred direction on scales of the order 500 particles. The merging of the first generation progenitors occurs over scales significantly larger than a grid spacing, thus removing concerns about a preferred collapse direction for these objects. It thus makes sense to create a small periodic glass with very low noise and then tile this within the simulation box.

To create the glass 'tile', 512 particles are placed in a periodic box. These particles are forced to be anti-correlated by not allowing any two particles to be closer than $0.9 \times N^{1/3}$, which is 0.9 times the average inter-particle spacing. This initial condition has a very low noise level. The noise level is further reduced by evolving the glass in a (periodic) gravity-only simulation with the sign of the velocity update reversed. With this modification, the particles seek to repel one another, and eventually relax to a state in which the (repuslive) potential energy is reduced to a minimum.

Once the tile is fully evolved, it is replicated a number of times to form the main simulation cube. This configuration does not have any noise on scales larger than the size of tile and thus constitutes an excellent initial configuration. The hierarchical layers are constructed by successively cutting out a region of the simulation cube and replacing it with a copy of the top-level 'grid' cut and shrunk to the appropriate size. The first layer, for example, is constructed by removing a sphere of radius 1/4 the box size and then filling that region with a sphere cut from the main simulation cube and shrunk to size. The next layer is formed by cutting a sphere of radius 1/8 the box and replacing this with a similarly cut and shrunk sphere from the main simulation cube.

Unfortunately this process does introduce some noise at the boundary of each region. It was thus assured that in the highest resolution region, objects of interest form sufficiently far away from the boundary with the next region. The layering process continues through four layers. To maintain mass resolution, the particles in each layer have mass 1/8 that of the previous layer. Thus the mass resolution of this region is 512 times higher than the lowest resolution region, and the spatial resolution is eight times higher. Figure 3.12 shows the layering in detail. If a grid of 32^3 particles is initially used to perform the layering, the resulting system has 77,813 dark matter particles and 17,165 SPH particles. In the high resolution region the effective resolution is 2×256^3.

Assigning the perturbations associated with the initial power spectrum is more difficult for multiple mass simulations. The particle Nyquist frequency is not constant across the simulation. If the

box is loaded with modes up to Nyquist frequency of the highest resolution region, then aliasing of the extra modes will occur in the low order modes of the low resolution region. Hence, to prevent aliasing, the lower resolution regions must have their displacements evaluated from a force grid that is calculated using only modes up to the local particle Nyquist. Thus all of the box modes are calculated, and then modes are progressively removed by applying a top-hat filter in k-space.

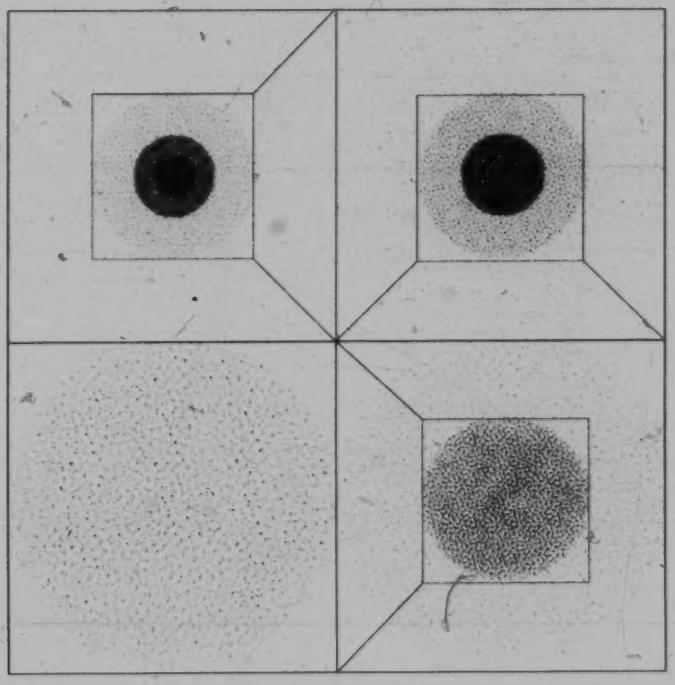

Figure 3.12: Layering of initial conditions. Starting clockwise from the upper left panel, the top level configuration of 32^3 particles is repeatedly cut and shrunk to create a hierarchy four levels deep. Gas particles are included in the central region only.

3.5.2 Simulation Parameters

To assign adiabatic gravitational perturbations, the linear CDM power spectrum of Bond and Efstathiou (1984) was utilized,

$$P(k) = \frac{Aq}{[1 + (23.1q + (11.4q)^{3/2} + (6.5q)^2)^{5/4}]^{8/5}}, \tag{3.8}$$

where,

$$q = k/(\Gamma h). \tag{3.9}$$

Given a baryon fraction of 10% the shape parameter, Γ, was calculated from Vianna and Liddle (1996) yielding $\Gamma \simeq 0.41$. The Hubble constant was set at 50 km s^{-1} Mpc^{-1}, yielding in $h = 0.5$ in the standard H_0=100h km s^{-1} Mpc^{-1} units. The normalization constant, A, was chosen so as to reproduce the number density of rich clusters as observed today, which is given by the rms mass variance $\sigma_8 = 0.6$ (Eke *et al.*, 1996). The initial starting redshift was $z = 67$ and the box size 50 Mpc (all length scales are quoted in real units).

To ensure the collisionless nature of the dark matter does not become contaminated by two-body forces, the two-body relaxation should be longer than the Hubble time. Thomas and Couchman (1992) show that for a uniform distribution of particles (of mass m, and softening ϵ) within a spherical volume of radius R and with a velocity dispersion $v^2 \sim \gamma GmN/R$ (γ is a constant dependent upon the characteristics of the velocity distribution) the two-body relaxation time is

$$t_r \simeq \frac{\gamma^{3/2} N^{1/2}}{6 \ln(R/\epsilon)} \left(\frac{R}{\epsilon}\right)^{3/2} t_2, \tag{3.10}$$

where $t_2 = (\epsilon^3/Gm)^{1/2}$ is the minimum time-scale for interactions of particles under gravity, with an effective impact parameter ϵ. Utilising the velocity dispersion for an isothermal sphere and modifying the t_2 parameterization of TC92, equation 3.10 may be rearranged to give

$$\frac{t_r}{t_0} \simeq 0.02 \frac{(R/\epsilon)^{3/2}}{ln(R/\epsilon)} N^{1/2} \left(\frac{\epsilon}{4\ \mathrm{kpc}} \frac{50\ \mathrm{Mpc}}{box}\right)^{3/2} \left(\frac{N_p^{eff}}{256^3 \Omega_p}\right)^{1/2} \tag{3.11}$$

Ideally the ratio t_r/t_0 should be greater than one, although since the simulation evolves to $z = 1$, values of 0.5 should be acceptable. The formula is minimized when $R = e^{1/2}\epsilon$ (*i.e.* close to the softening), and at this radius it is found that $t_r/t_0 \simeq 0.08N^{1/2}$. This implies that within the softening volume, $N > 10$ is neccesary to avoid two-body relaxation, although if the simulations were integrated to $z = 0$, the criterion would be $N > 30$.

The effective resolution of the high resolution region (which is 6.25 Mpc in diameter) is 256^3, which yields a mass resolution of 4.6×10^8 $M_\odot$ in the dark matter, 5.1×10^7 $M_\odot$ in the gas (reducing to 2.6×10^7 $M_\odot$ after the creation of the first star particle) and 2.6×10^7 $M_\odot$ in the star particles. Clearly the mass resolution remains low, with a 10^{11} $M_\odot$ galaxy (in baryons) being represented by approximately 4,000 gas and star particles, assuming an equal division of both. Nonetheless, this resolution is sufficient to give a reasonable indication of the performance of different algorithms in a cosmological environment. The total baryonic mass in the high resolution region is approximately 8×10^{11} $M_\odot$. This is a consequence of attempting to keep the boundary of the second mass hierarchy a sufficiently long way from the object of interest, and choosing a sufficiently large box size to provide a reasonable representation of tidal forces. Due to the low resolution of the simulation, shear-correction was applied to the artificial viscosity.

The first simulation conducted was a low resolution 128^3 dark matter simulation using the parameters given. From this simulation, candidate halos for re-simulation were extracted at a redshift of $z = 1$. Due to wall-clock limits on simulation time, it was decided that $z = 1$ was the most appropriate time to stop the simulation. Whilst the dark matter run could have be continued to $z = 0$, this is not possible for the high resolution hydrodynamic runs. The chosen halo had a mass of 2.7×10^{12} $M_\odot$ and is thus comparatively large. Re-simulation showed that it corresponds to the halo of a merger event of two galaxies with a combined bayonic mass of 1.8×10^{11} $M_\odot$.

3.5.3 System evolution without feedback

Without feedback, the system follows the ubiquitous cooling catastrophe picture. Baryons condense in the halos and radiatively cool rapidly due to their high density. A disc galaxy is formed in the center of the high resolution region, with a (baryonic) mass of 1.2×10^{10} $M_\odot$. The disc exhibits a (visibly striking) cutoff in particle density at a radius of 8 kpc, and similarly, the vertical distribution of the disc falls off abruptly above and below 1.5 kpc of the equator. A double exponential fit of the gas density profile (see section 3.5.6) clearly displays the rapid fall-off in density with radius beyond 8 kpc. Star formation proceeds rapidly due to the high density, and is initially concentrated in the nucleus of the disc (which contains 40% of the baryonic mass and has a 0.6 kpc diameter—much smaller than the softening length). Stars formed in the nucleus diffuse away from it, forming a stellar bulge approximately 3.0 kpc in diameter (compare the radial density profiles in Figure 3.15). Due to the low resolution, the hierarchical formation of the disc is represented poorly, with only a handful of progenitors merging to form the disc.

At late times $z = 1.09$, the disc exhibits a number of features that have been observed previously. There is a deficit in angular momentum, with the specific angular momentum of the baryons corresponding to that of an elliptical system for the given mass scale. Consequently the disc does not exhibit a large radial extent. At $z = 1.01$ the disc undergoes a major merger with another system (of mass 6×10^{10} $M_\odot$ (in baryons) at a speed of 300 km s^{-1} relative to the center of mass frame for the major disc). As is generally observed in simulations with stellar and gaseous components, the

Run	feedback[a]	R_{200} (kpc)	Σm_* (10^{10} $M_\odot$)	$\lvert L_{gc}\rvert/\lvert L_{dm}\rvert$	M_{disc} (10^{11} $M_\odot$)	R^b_{disc} (kpc)	R^c_{inner} (kpc)	R_{outer} (kpc)
6001	SPa	187	7.98	0.25	1.07	8.8	1.0	34
6002	SPna	187	8.44	0.24	1.14	7.6	0.7	30
6003	ESa	188	9.51	0.09	1.38	8.0	0.6	17
6004	ESna	188	9.38	0.18	1.42	11.3[†]	0.7	24
6005	TSna	189	7.45	0.19	1.35	9.3	0.8	27
6006	TSa	188	7.25	0.27	1.27	9.3	0.9	24
6007	NF	189	9.67	0.16	1.57	9.5	0.9	34
6008	NSF	188	N/A	0.14	1.29	9.3	0.6	28
6010	TS	188	8.69	0.19	1.38	9.7	0.6	21
6014	ESa-SG	189	9.78	0.18	1.25	9.6	1.1	45
6015	ESa-2c*	189	10.9	0.15	1.24	9.1	0.9	50
6016	TSa-SG-2c*	188	6.33	0.33	1.06	10.3	1.3	39
6017	ESa-SG-2c*	189	11.6	0.08	1.35	9.2	0.8	36
6018	ESa-nav	188	3.83	0.23	1.34	8.2	0.9	36

Table 3.3: Summary of the properties of cosmological simulations at z=1.09. [a]SPa=Single particle adiabatic period, SPna=single particle no adiabatic period but adjusted cooling density, ESna=Total energy smoothing with adjusted cooling density but no adiabatic period, ESa=Total energy smoothing with adiabatic period, TS=Temperature smoothing (normal cooling). [b]The disc radius was evaluated by finding the radius at which the baryon surface density fell below 2×10^{13} $M_\odot$ Mpc^{-2}. This value was established by visually judging the edge of the NSF disc and then reading off the surface density at this boundary. [c]The inner values are distorted to shorter scale lengths by the presence, or lack of, a central core in the disc. The outer fits are strongly affected by companion systems inside r_{200} and strong emphasis should not be placed on these results. [†]This value is anomalously high due to a minor merger during which the stellar content of the merging dwarf orbits out at > 10 kpc.

resulting morphologies of the gas and stars differ significantly. The gas cores merge, creating a very dense core while the stellar components merge, producing 'shells' as observed in ellipitical galaxies. A tidal tail is also produced during the merger and is populated by both gas and stars.

3.5.4 Caveats

Unfortunately, even though the SFR normalization was adjusted to 0.025, the SFR in the simulations appears to be somewhat low. Although the plots in Figure 3.13 show SFRs in excess of 70 $M_\odot yr^{-1}$, it should be noted that this value is integrated over 8×10^{11} $M_\odot$. Diagnostics from the simulation indicate that of this mass, $6.5 - 7 \times 10^{11}$ $M_\odot$ is not involved in star formation ($T > 30000$ K or $\rho < \rho_{sf}$). Beyond the main disc and the merger companion (a combined baryonic mass of 1.8×10^{11} $M_\odot$ of which 60% is in star forming regions), tertiary halos contribute only 2.4×10^{10} $M_\odot$ of star forming matter. Hence, the bulk of the SFR is derived from the main disc and its major companion. It should be emphasized that the halo correspondence between simulations is not perfect, but given the difficulty in accurately calculating the cooling rates at low resolution, this is not surprising. Well-defined halos, *i.e.* those formed with 500 or more particles, do correspond well, as can be seen in the radial plot in Figure 3.19. There are also small synchronization errors (10^5 years) between the analysed time-step outputs. To examine the effect of changing parameters a number of auxiliary simulations were ran, the details of which are discussed in section 3.5.8.

3.5.5 Effect of feedback on SFR and morphology

The most noticeable difference in the ensemble is that the temperature smoothing version does not lead to a significantly different final structure. This is contradictory to the isolated results where temperature smoothing is seen to promote violent winds and disc disruption. The reason for this is that the density of the first objects is so high, $n_H > 1$ cm^{-3}, that the cooling time ($\simeq 0.1$ Myr) is short enough to remove the SN energy within a time-step, unlike the isolated simulation where the

resolution is high enough to allow a reduction in density due to expansion of the feedback region, consequently leading to a reduction in the cooling time. This is simply the cooling catastrophe. To test what happens when the feedback energy is allowed to persist, the TS simulations were ran with the adjusted cooling mechanisms. As expected, these simulations produced more diffuse structures due to the large amount of energy input and also the allowed persistence of feedback regions.

At z=1.09, the morphology of the major disc was examined. Without exception, all the simulations produced a disc with a clearly defined cutoff radius of $9.2^{+2.1}_{-1.6}$ kpc ($> 2h_{min}$). This result is the *same as the run with no feedback*. However, models including feedback were fatter at the disc edge, (the TSa run with a thickness of 5 kpc, being 30% wider than the NSF run). 'Bubbles' were noticeable in the discs, more so in the ESa run than others because the TSa and TSna runs were already quite diffuse. Feedback did not change the radial extent of the disc which suggests that it must be determined largely by the dark matter potential. It cannot be due to the central concentration of baryons, since the TSa and TSna runs effectively destroy this concentration yet still have approximately the same radius.

While the internal structure of the major disc was not significantly different across all simulations, that of the merged system was. For the no feedback simulation, the gaseous cores were much more tightly bound than for the TSa and TSna, but largely similar to the ESa, Esna, SPa and SPna runs. In particular, because the gas cores are sufficiently inflated in the TSa and TSna runs, the gas undergoes a smooth merger, and for the TSa run the feedback is sufficient to produce a *disc* as the result of the merger. Note that the stellar components evolve in a similar fashion though, producing shells, and a widely dispersed final stellar structure.

The SFRs for the main simulations are plotted in Figure 3.13 . The upper left panel shows the results for the simulation without feedback, and gives an illustration of the smoothing effect of the 160 step running average used to smooth the data. All algorithms agree on the early SFR, which reaches 1 $M_\odot$ yr^{-1} at $z = 3.9$, since sufficient time must pass for the star mass of particles in the highest density regions to reach the mass threshold for creating a star particle (the first star particles are created at $z \simeq 3$). At late times, the merger causes a strong star burst which is visible in all of the SFRs, albeit somewhat suppressed in the simulations with strong feedback. The relative effect of the different feedback algorithms was compared by calculating the reduction in the cumulative mass of stars at $z = 1.09$, as a percentage relative to the no feedback run (total 9.7×10^{10} $M_\odot$). The algorithms with the most significant effects are, in order, TSa (25%),TSna (23%), SPa(18%) SPna (13%) and TS(10%) while the energy smoothing variants ESna (3%), ESa(2%) have comparatively little effect on the SFR. As in the isolated simulations, the SP algorithms eject particles due to asymmetry in the local distribution of particles and the subsequent reduction in the gas density is the main source of the SFR reduction over the energy smoothing variants.

3.5.6 Halo profiles

In view of the results from the isolated simulations, the halo structure was examined to see if there was any difference between algorithms. Figure 3.14 compares the gas halo temperature for the NF, ESa and SPa runs. The higher temperature seen at the edge of the SP profile (beyond 200 pc), is difficult to attribute just to 'hot' particles being ejected to that radii. Tracing the orbits of a number of ejected particles showed that the largest distance they achieve from the core is 150 kpc. It is noticeable that at a radius of 150 kpc, the temperature of the SP feedback halos is higher than that of the others. A plot of the radial pressure showed that beyond 200 kpc, the pressure in the SP halos is a factor of two higher than in the other runs. Also a plot of the cumulative density versus radius confirms that more of the gas lies at large radii (beyond 200 kpc) for the SP feedback. This indicates that the SP feedback is producing a form of pressure support for the halo which is in turn leading to higher temperatures in the infalling matter at large radii (since the gravitational compression remains dominated by the dark matter).

The density profiles for the baryons can be fit by a double exponential, with the break between the two profiles occurring at the edge of the disc. An argument can be made that the presence of the gas/stellar core suggests that the disc should also exhibit a double profile. However the structure is sub-resolution. In particular, when the smoothed density is examined (which is used in the SFR calculation), there is very little difference between simulations. A summary of least squares fits for the inner and outer parts of the density profiles is given in table 3.3. The fitting was somewhat arbitrary since the break between the fits is decided by eye. Note that for the TS variants, this was particularly difficult since the transition from disc to halo is less clear, i.e. the density curve is smoothly decreasing as opposed to a sharp discontinuity visible in the other data sets. The inner fits, which give an effective scale length for 3-dimensional baryon distribution in and about the disc, are broadly similar and are given by $s_L = 0.75^{+0.25}_{-0.16}$ kpc (ignoring the auxiliary simulations). Note that the gas cores tend to distort the fits toward shorter scale lengths and this three-dimensional profile underestimates the scale length that would be interpreted from a surface density plot. There

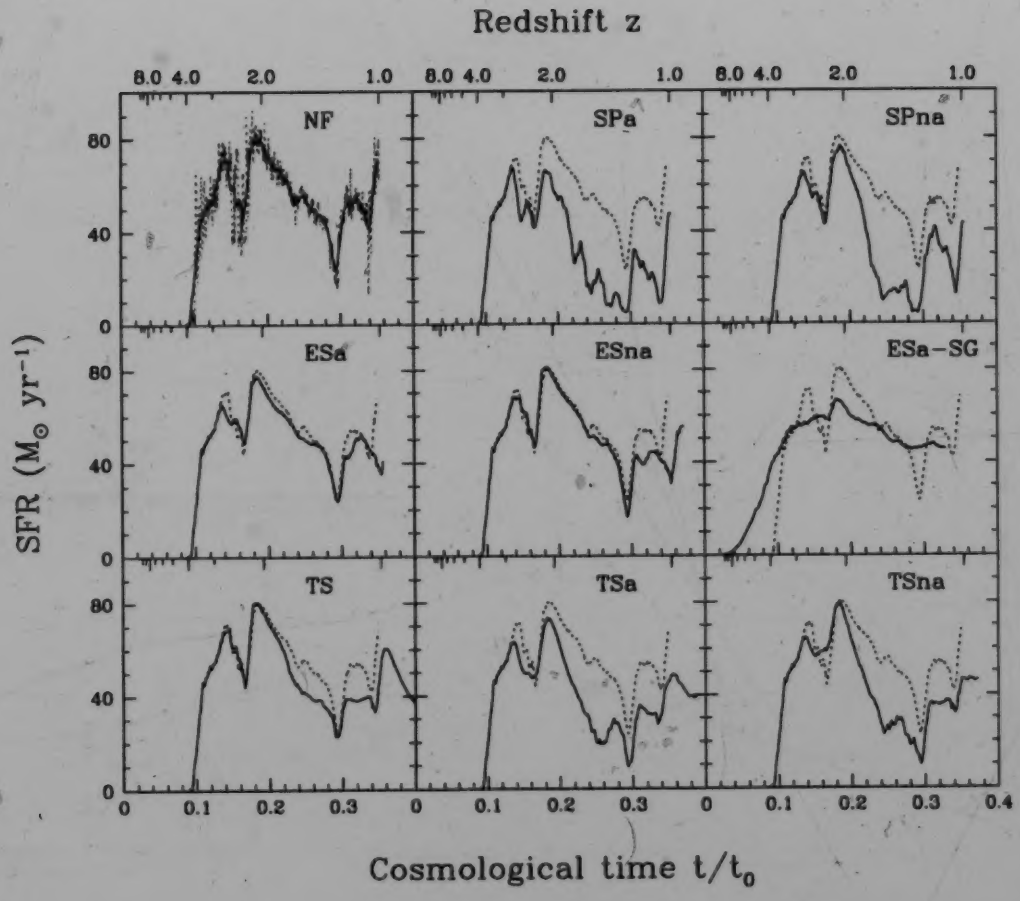

Figure 3.13: SFRs for the cosmological simulations. The SFR shown is integrated over the entire gas sector of the simulation (8×10^{11} $M_\odot$). A 160 time-step average is used to smooth the data and more clearly elucidate trends the effect of the smoothing is demonstrated in the no feedback panel. For comparison, the NF SFR is plotted as a dotted line in the remainder of the panels.

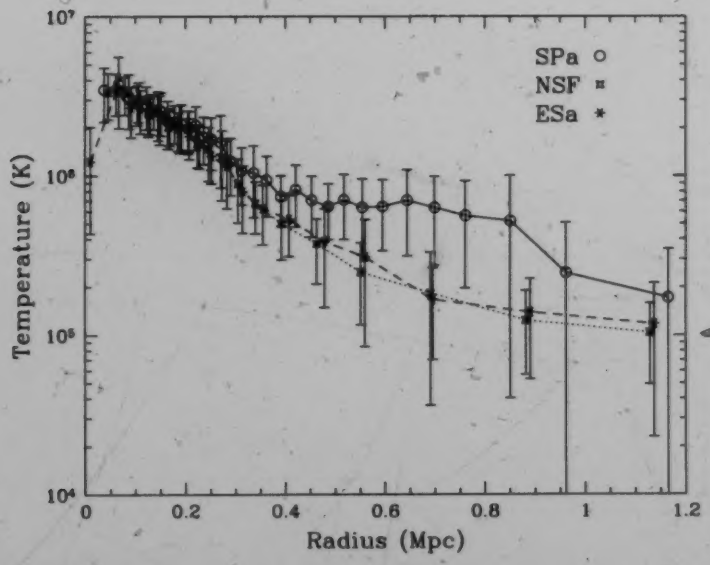

Figure 3.14: Radial temperature profile for the NF, SPa and ESa runs. The error bars denote 1σ bins about the mean, with the data plotted in Lagrangian bins of size 208 particles. The SPa and SPna (not shown) profiles both exhibit a higher temperature at large radii (see text). Of the remaining algorithms, all follow profiles similar to the ESa and NF simulations.

is a clear trend that the adiabatic feedback schemes have longer scale lengths that the non-adiabatic versions. This is to be expected since the adiabatic feedback keeps the gas more diffuse. The outer fits are more problematical since satellites distort the radially averaged density profile quite severely. Given the sensitivity of the slope to these perturbations, it is difficult to draw any conclusions from these data.

The gas and stellar density profiles are broadly similar since the stellar disc evolves out of the gas. Star particles that are formed within the dense central gas core eventually orbit at a larger radii than the parent particle since they are not affected by the viscous forces felt by the gas. A comparison of the gas to stellar density profile is shown in Figure 3.15. The smoothed gas density (*i.e.* the SPH density) is remarkably similar across all simulations indicating that the SFR should be similar (modulo the effect of feedback events). The clear rise in the density at small radii is a signal of the gas core, albeit at sub-resolution scales. For the TSa/na runs this core was removed due to the strong feedback. For the SP runs, the ejection of particles also lowered the core mass. The ES runs were incapable of inflating the core once formed, showing that dense cores are particularly difficult to inflate once formed.

The density profiles for the dark matter differ little from simulation to simulation (r_{200} differs across all simulations by only 1%). At least with this mass resolution, there is no evidence for feedback being capable of rearranging the dark matter structure (Navarro *et al.*, 1996*b*), especially in view of its inability to affect the baryonic structure.

3.5.7 Rotation curves and angular momentum

Figure 3.16 displays the (Plummer softened) rotation curves,

$$v_c^2(r) = \frac{GMr^2}{(r^2+\epsilon^2)^{3/2}}, \tag{3.12}$$

and particle tangential velocities for the run without star formation compared to the ESa, SPa and TSa runs. For all simulations, the tangential velocities rise more sharply than the rotation curve. However, the initial slope of the rotation curve is dominated by the softening parameter and a 12%

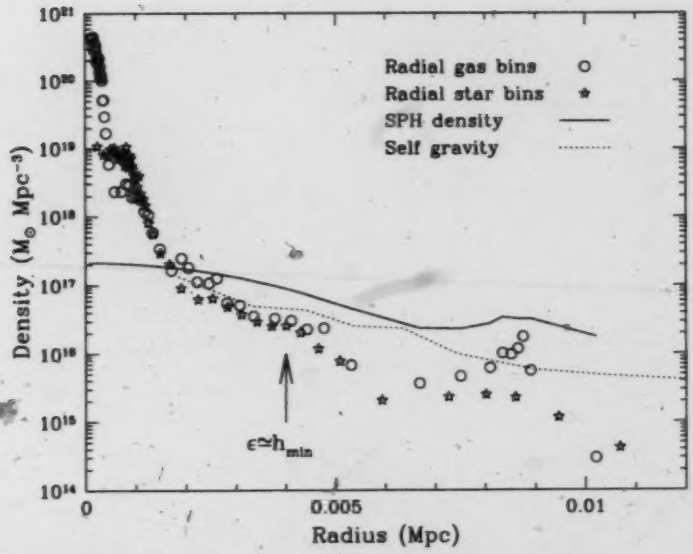

Figure 3.15: Density profile for the NF run. Spherical Lagrangian bins (52 particles) have been used to bin the star and gas data. Clearly the gas and stellar data are comparatively similar. The stellar bulge has roughly constant density and extends to a radius of 1.5 kpc, the gas nucleus extends only just beyond 0.5 kpc. The SPH density data is a radial binning of the raw SPH density values which, because of smoothing, do not increase to the exceptionally high values seen in the radially binned stars and gas. Further, it reflects the 2-dimensional density better than the spherical bins. The self-gravity line corresponds to 0.4 times the spherically binned dark matter density.

reduction in the softening length is enough to fit the tangential data, hence this should not be considered a significant discrepancy. Also note that a slight limitation in the grid solver (to prevent the grid force becoming too 'soft') means that at early epochs, a shorter softening length than desired is used. This artifact may be reflected in the results. The 300 km s^{-1} peak of the rotation curve is consistent with the mass of the disc and halo, although the SPa run is slightly lower since gas has been ejected out of the disc into the halo.

The TSa run shows significantly increased dispersion in the tangential velocities because of the feedback, but does not have a larger disc diameter (in keeping with the similar scale lengths found in the analysis of the density profiles). Since particles involved in feedback regions tend to be ejected vertically from the disc, *i.e.* preferentially into low density regions, a large tangential velocity dispersion can arise. This is seen most clearly in the TSa a plot.

In view of the rotation curves being similar across all simulations, it would be expected that the velocity dispersions should also exhibit similar profiles. A comparison plot of the NSF run compared to the SPa, TSa and TSa-SG-2c* run (note this run has double the fiducial SFR and no self-gravity criterion, see section 3.5.8) is shown in Figure 3.17. As expected, the profiles are broadly similar with a maximum difference between the plotted runs of 20 km s^{-1} at the disc edge. It is interesting to note that the TSa run has a large tangential velocity dispersion, as is visible in Figure 3.16, but a comparatively low dispersion in the radial direction. This is a result of feedback preferentially ejecting particles vertically, in turn boosting the tangential velocity component more than the radial. The data for NF run (not shown) are dominated by an ongoing merger which introduces a very large dispersion (120 km s^{-1}) at the edge of the disc, far larger than that produced by any feedback. The NF and ESa runs also show the effect of the merger, with peaks in the data at 4.2 kpc and 3.6 kpc respectively corresponding to the position of the strongest perturbation within the disc. The SPa and TSa runs are less affected by the merger since the dwarf has been reduced in mass by the stronger feedback. It is clear that the low resolution in the discs and the complications of ongoing mergers make it difficult to draw conclusions from this data.

To see how much angular momentum has been lost by the disc, the specific angular momentum of

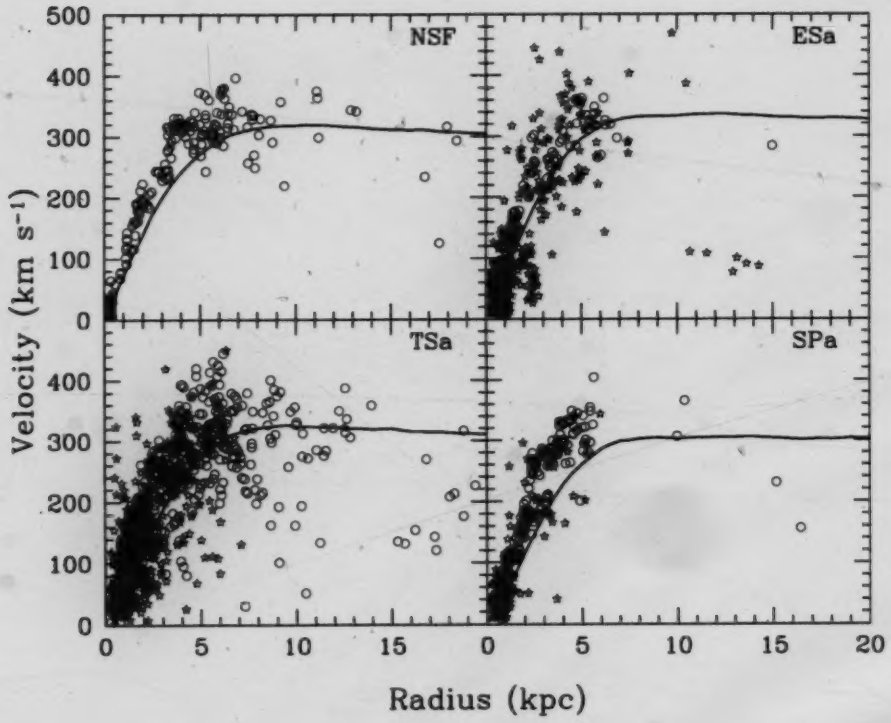

Figure 3.16: Rotation cruves for the NSF, TSa, ESa and SPa runs. The solid line shows the Plummer softened rotation curve while points indicate the tangential velocity of individual particles. Circles represent gas particles and open stars stellar particles. Particles plotted lie in a band 10 degrees wide about the plane perpendicular to the angular momentum vector of the gas. More particles appear in the TSa plot since the dense stellar core is inflated while in the other plots most of these particles fall outside the selected plane (since they are contained in the dense core). The integrated rotation curves are broadly similar and the particle data differ only marginally, with more velocity dispersion being visible in the TSa run.

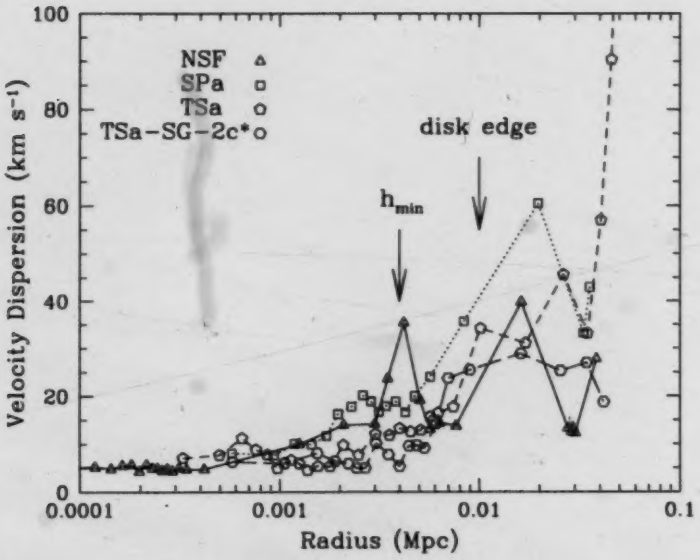

Figure 3.17: Radial velocity dispersions in the gas disc. The minimum smoothing length h_{min} and disc edges are marked for clarity. In contrast to the isolated simulations, there appears to be no correlation between more violent feedback producing higher velocity dispersion. This is partially due to the fact that merging dwarfs produce a far higher velocity dispersion than feedback, and also the discs are not well resolved.

the cores is compared to the dark matter halo (within r_{200}) in Figure 3.18. The angular momentum for the halo gas (the gas for which $\delta < 2000$ within r_{200}) and that of the stellar component of the disc are also shown. For all simulations, the disc system shows a deficit of specific angular momentum when compared to the dark matter. By breaking the disc into its stellar component and gas component, it becomes clear that in the simulations with feedback that are shown (TSa,SPa and TSa-SG-2c*) there is a *trend toward higher angular momentum values for the gas disc.* The highest value, that from the TSa-SG-2c* simulation, just falls within the disc region of the parameter space. However, the stellar discs all fall in the elliptical region of the parameter space, and the effect of feedback on them is small. Note that the purely gaseous run also sits in the elliptical region and the gas disc in the no feedback run has marginally higher specific angular momentum than the stellar component. The NF run is misleading, since a merger is going on at the edge of the disc leading to higher angular momentum values as compared to the other feedback runs (although by the end of the merger the opposite will be true due to core-halo interaction). For all simulations, the angular momentum of the halo gas is larger than that of the dark matter since the dark matter plot includes the contribution of the dark matter core, which has little angular momentum but a significant amount of mass. It is clear from the plot that if the halo gas were to fall smoothly onto the disc, then it should be possible to form a disc with an angular momentum value midway between that of the halo gas and disc system. Note infalling satellites may still disrupt the disc and cause still further angular momentum loss.

In Figure 3.19 the z-component of the specific angular momentum **L** is shown. If significant angular momentum loss occurs as a result of disc formation then a larger proportion of the disc mass should lie beneath the line formed by Rv_c. Since this is not the case, it is clear that at $z = 1.09$ there has been little angular momentum loss (within the disc) due to bar formation. However, all the discs are deficient in angular momentum relative to the halo, since they have been formed in a hierarchical process which is subject to core-halo angular momentum transport. Remarkably, the $X_2(R)$ stability parameter plots are all similar, with all the discs achieving the $X_2(R) = 3$ stability requirement just beyond the 4 kpc softening radius. As shown in DT99, it is more than likely that if the baryonic mass is redistributed into an exponential disc, then the $X_2(R)$ parameter for this system does not achieve stability until a much larger radius, *i.e.* the cores provide disc stability. Note that the kink in the $X_2(R)$ plot for the NF run is due to the merger previously discussed and

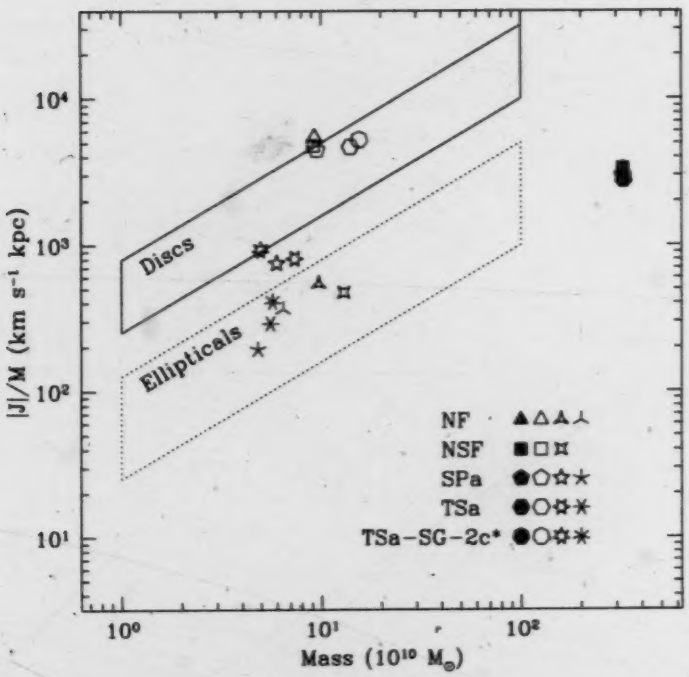

Figure 3.18: Specific angular momenta versus mass for different components of the system for a number of different feedback algorithms. The filled polygons plot the angular momentum of the dark matter within r_{200}, the open polygons that of halo gas (all gas that does not fall above $\delta = 2000$), pointed stars that of the gas in the main disc ($\delta > 2000$) and finally the centrally connected stars show that angular momentum of the stellar component of the disc. The runs with feedback show a small but noticeable trend toward higher angular momentum values for the gas disc component (contrast with the NF run). Both dark matter and gas halo values are in broad agreement as expected.

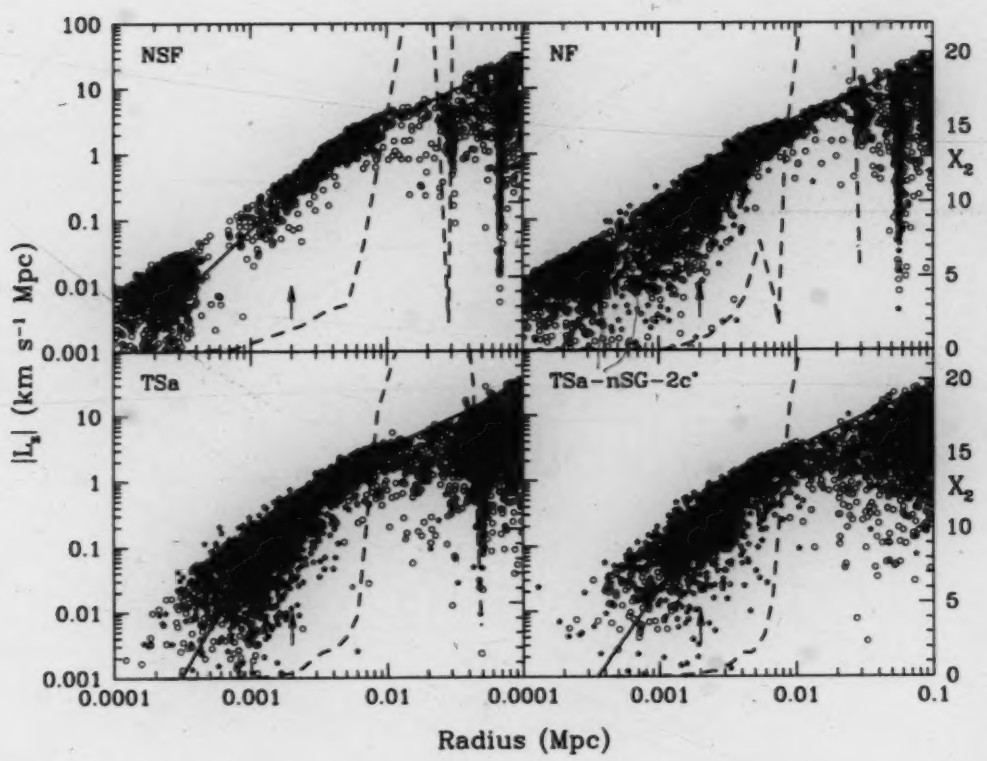

Figure 3.19: Plot of specific angular momentum versus radius, showing the raw particle data versus $|L_z|$ calculated from the rotation curve at $z = 1.09$ (counter rotating particles are not shown). The $X_2(R)$ stability parameters for the discs are also shown. The raw data fits the expected curve well, indicating that at least at the current epoch (prior to significant accretion) there has not been significant angular momentum loss as a result of disc formation (a bar is yet to form). The large concentration of matter at very small radii in the NF and NSF runs is the central core and is slightly offset, since the center of mass velocity was measured over the entire disc rather than the core. The $X_2(R)$ plots are all similar with little difference even for the extreme feedback in the TSa-SG-2c* run. Disc stability occurs at $X_2(R) = 3$ and is achieved at the smallest radius of 3.5 kpc in the NSF run and at the largest radius of 6.0 kpc for the TSa-SG-2c*. Note that the TSa runs provide sufficient feedback to remove the central mass concentration, but fail to increase the disc radius significantly.

it does not affect the disc stability greatly (the kink drops to 2 at 6.5 kpc but quickly returns to stable values).

3.5.8 Auxiliary simulations

To adequately examine the parameter space of these simulations and also determine the effect of SPH algorithm changes would take an excessively long time. Alternatively, by conducting a few auxiliary simulations, much can be learnt about the outer edges of the parameter space. To understand what happens when the SPH algorithm is changed, in particular in relation to the treatment of high density regions, it is simple to contrast to one of the previous simulations run with the same parameters.

As indicated in previous sections, to determine the effect of removing the self-gravity criterion, the ESa simulation was rerun without it. This simulation is denoted ESa-SG. Also the effect of doubling the star formation rate normalization was tested in a simulation denoted ESa-2c*. Since most of the simulations conducted appeared to be relatively unchanged by the introduction of feedback, a simulation with extremely violent star formation and feedback was run (twice the fudicial SFR normalization with TSa feedback and also without the self-gravity criterion). This simulation, denoted TSa-SG-2c*, is not particularly realistic (star formation begins very early and the feedback is over-efficient) but it does provide an excellent guide to the limits of what feedback can accomplish. Because of the strong feedback, the formation dense gas cores was all but prevented, with the exception of very large $> 5 \times 10^{10}$ $M_\odot$ systems. Hence it was possible to integrate the system to later times since the SPH algorithm did not suffer significant slow down. An ESa-SG-2c* simulation was also ran to contrast with the temperature smoothing version.

As an alternative to smoothing over all the particles within the minimum smoothing length, h_{min}, a number of authors continue to search over only N_{smooth} neighbour particles (*e.g.* Navarro and White, 1993; Bate and Burkert, 1997). Such a procedure places a direct limit on the maximum resolved density: the volume normalization is set by h_{min} and the summation over neigbours is limited to N_{smooth} particles. In turn, this sets a bound on the maximum SFR per particle and consequently within the system as a whole. It also sets a bound on the cooling rate. The reasons for making this change are primarily related to efficiency: if a simulation smooths over all the particles within h_{min} it can be exhibit a severe slowdown. For example, nearly all the simulations without this adjustment slow down by a factor of 7 from start to finish. Hence to examine the effect of making this change, the ESa simulation was re-run with this high density treatment. This simulation is denoted ESa-nav.

In the ESa-SG run, the effect of removing the self-gravity criterion is that star formation begins at a very early epoch (1 $M_\odot$ yr^{-1} at $z = 8.3$). It is initially confined to a few particles (recall the $\nabla.\mathbf{v}$ criterion must also be fulfilled) leading to an SFR of 0.2 $M_\odot$ yr^{-1}. During later evolution, the SFR is only marginally lower than that of the ESa run, the largest difference being 10 $M_\odot$ yr^{-1} at $z = 2.1$ (a 12% difference). Further, the disc and halo structure are comparatively similar, as measured by density and temperature profiles, and the dense gas core is still formed.

In Figure 3.20 the plot of the SFR in the ESa-2c* simulation shows that doubling the SFR normalization leads to a stronger initial star burst as the gas overcomes the self-gravity criterion, but does not produce an SFR that is exactly double that of the standard simulations. The peak SFR of 120 $M_\odot$ yr^{-1} at $z = 2.1$ is 56% higher than that in the ESa run (77 $M_\odot$ yr^{-1}). The SFR is not simply doubled because of both increased feedback and the finite amount of gas available for star formation. Notably the dense gas core was still formed, showing that even with the increased SFR, ESa is still incapable of producing a significant effect on morphology.

As expected, the TSa-SG-2c* simulation leads to markedly different results than any of the previous simulations. Because of the extreme feedback and consequent absence of small progenitors, the disc assembly process is very smooth. There is essentially no formation of a gas and stellar nucleus within the disc. Star formation begins at the same epoch as the ESa-SG run, albeit at a higher rate due to the increased SFR normalization which yields 1 $M_\odot$ yr^{-1} at $z = 9.1$. At $z = 6.2$ the SFR reached 15 $M_\odot$ yr^{-1} and a peak of 52 $M_\odot$ yr^{-1} occurred at $z = 3.0$. The most noticeable difference in the SFR is that at late times it falls off precipitously. At $z = 1.0$ the SFR is 18 $M_\odot$ yr^{-1} compared to 36 $M_\odot$ yr^{-1} for the ESa run and by $z = 0.5$ it had fallen to 5 $M_\odot$ yr^{-1}. Note the decay in the SFR versus time was linear rather than exponential, This has been observed before in parameter space explorations (Thacker, 1997). The less energetic feedback provided by energy smoothing lead to the ESa-SG-2c* simulation producing a much higher peak SFR, namely 80 $M_\odot$ yr^{-1} at $z = 2.7$. In keeping with all the other energy smoothing simulations, a central gas nucleus was formed in the disc and the temperature and density profiles remain similar to the ESa run.

At $z = 0.5$, the TSa-SG-2c* disc was analysed to see if the accretion of the halo gas had indeed allowed the formation of a disc without an angular momentum deficit. By this epoch, the disc had

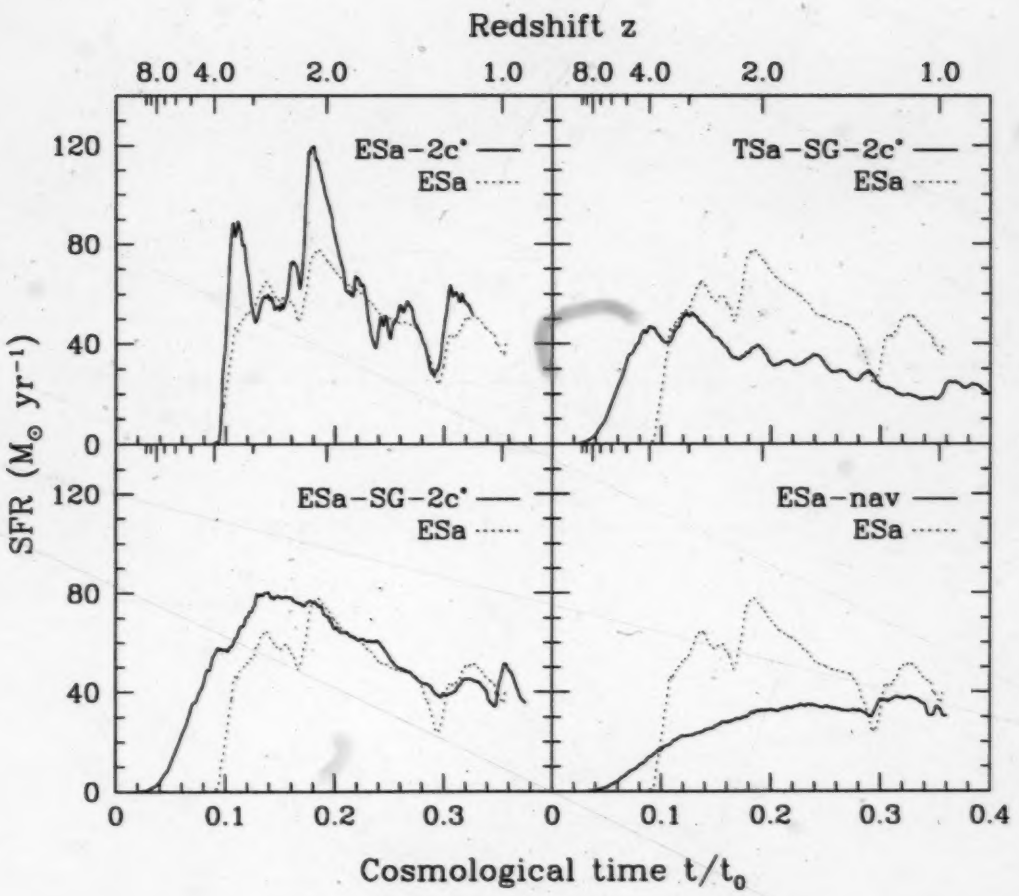

Figure 3.20: SFRs for the auxiliary cosmological simulations. The SFR shown is integrated over the entire gas sector of the simulation (8×10^{11} $M_\odot$), and time averaging is used to smooth the data. Doubling the SFR normalization does not lead to a perfect doubling of the SFR, although peak values are close. Removal of the self-gravity criterion does lead to the very early on set of star formation. The TSa-SG-2c* simulation has strong feedback and even with the higher SFR normalization still has a lower peak SFR than the ESa run (plotted for comparison). The ESa-SG-2c* run, while having a peak SFR earlier, is not significantly different from the ESa run. Changing the short range behaviour of the SPH solver leads to a markedly different SFR, as shown in the plot of the ESa-nav data. The peak of the SFR is later and lower, and the self-gravity criterion only mildly delays the onset of star formation.

grown to a diameter of 12.0 kpc, which is 17% larger than the value from $z = 1.09$ and the mass had increased by 30% to 1.39×10^{11} $M_{\odot}$. r_{200} had grown to 285 kpc. The ratio of the specific angular momenta for the gas core and dark matter, increased to 0.40 (20% increase). A visualization of the system shows that the ratio cannot increase significantly as there are very few cold gas clumps within the dark matter halo available for accretion. Most have been blown apart by feedback. Note that at the center of the halo the gas density is approximately $n_H \simeq 10^{-3}$ cm$^{-}3$ and the temperature is over 10^6 K, leading to a cooling time of greater than 4×10^9 years. Hence very little of this gas, which has a large specific angular momentum, can cool on to the gas disc before $z = 0$.

The results from the ESa-nav simulation are very different. The peak SFR is 38 $M_{\odot}$ yr^{-1} and it occurs at a much later time than the rest of the simulations ($z = 1.1$). Also star formation begins slightly later than that seen in the simulations without the self-gravity criterion (1 $M_{\odot}$ yr^{-1} by $z = 6.9$) which is significantly earlier than the standard simulations (1 $M_{\odot}$ yr^{-1} by $z = 3.9$). The reduced SFR is due to the density values calculated by this method being lower and not due to any increased effect of feedback. The change in the epoch at which the self-gravity criterion is overcome is due to the change in the search radius. In the center of the halo the dark matter profiles have a much shallower density profile than the baryon cores. When the neighbour search is conducted over the reduced radius only the core is sampled rather than the full 4 kpc radius, *i.e.* the sharp decline in baryon density at the disc edge is ignored. The disc formed is broadly similar to that in the ESa run, although it has a slightly larger baryon core, a thinner structure (2 kpc thick) and a smaller radius.

As was expected, the wall-clock per time-step slowdown was less severe for this code and it was roughly double the speed of the standard implementation. Note that this is an underestimate of the efficiency since the h-update algorithm did not accurately calculate the required change in the search radius for regions where it was less than $2h_{min}$. The algorithm preferentially smooths over too many particles, and an accurate calculation would require a very short time-step.

3.6 Summary and Discussion

This chapter details a study of a number of different feedback algorithms, comparing the effect on high resolution isolated systems and low resolution hierarchical simulations. The parameter space of the model was explored as were the effects of small changes in the hydrodynamic solver.

Principal conclusions follow.

1. As would be expected on energy budget grounds, the temperature smoothing (TS) feedback algorithm has the most impact on structure formation. Single particle (SP) feedback has a less significant but still noticeable effect and it produces a distinct change in the temperature profile of the halo at large radii. Energy smoothing (ES) is the least effective of the three fundamental mechanisms.

2. Feedback can be shown to have a large effect in systems that are well resolved in terms of particle number. In particular, the Milky Way and NGC 6503 prototypes are strongly affected by feedback even though the softening lengths were chosen to be comparable with those of cosmological simulations. Both 'blow-out' and 'blow-away' can be produced, depending upon the feedback algorithm.

3. The NGC 6503 prototype is more strongly affected by feedback than the Milky Way prototype, *i.e.* feedback has more effect on low mass systems, which confirms expectations. In the NGC 6503 model it is possible to differentiate differences between the cooling mechanisms.

4. In hierarchical simulations even an excessive amount of feedback, TSa-SG-2c* for example, produces little effect on the properties of large ($> 10^{11}$ $M_{\odot}$) disc galaxies at early epochs. Rotation curves and density profiles remain broadly similar.

5. Although small at the mass scale probed by the hierarchical simulations, there is a distinct trend toward higher specific angular momentum values in the disc with increased feedback. At lower mass scales, equivalently at higher resolution, the effect should be more noticeable.

6. The revised cooling mechanism has little effect for the energy smoothing algorithm since insufficient energy is input, and consequently, the estimated density does not lower the cooling rate significantly. Conversely, for single particle feedback and temperature smoothing, the energy input is more than sufficient to force the cooling rate from the estimated density to be much longer than the local one.

7. Morphologies remain broadly similar in hierachical simulations although feedback can reduce the SFR by as much as 30 $M_{\odot}$ yr^{-1}. Note that the reduced SFRs continue to offer similar peaks and troughs albeit at a lower overall normalization.

8. The introduction of a self-gravity criterion to prevent catastrophic star formation at high redshifts does not lead to significantly different structure formation provided that the SFR normalisation is set within reasonable bounds.

9. The treatment of the high density end of the SPH solver can produce an enormous difference in the SFR. By reducing the neighbour search in the high-density regions the SPH density becomes lower and consequently so does the SFR. Also the self-gravity criterion is overcome earlier since as the search radius is reduced the baryonic cores have a higher weighting.

At the resolution provided by the cosmological simulations hierarchical merging is not well defined. Only a handful of progenitor halos merger to form the major disc system. That all of the simulations, including those with the exceptional feedback provided by temperature smoothing, produce an over-compact disc is not cause for concern—the radius of the disc is determined largely by the depth of the potential well. Due to the early stage at which the simulations were stopped, it is difficult to comment on the evolution of the majority of disc systems, which are expected to accrete a large fraction of mass from $z \simeq 1$ onwards. The one simulation that was integrated to $z = 0.5$ gave some surprising results. If feedback is sufficiently strong to reduce the angular momentum problem for the largest systems then there are insufficient halos at later times to accrete on to the disc. Although it is a significant jump to go from this result to the idea that feedback was stronger at higher redshifts it certainly seems appealing. Silk (1998) discusses variations in the IMF over time.

One particularly important area that has not been examined is the effect of resolution. For star formation algorithms based upon density values, there are many issues to be considered. In particular, in a hierarchical cosmology, star formation will be higher and begin at earlier epochs with increased resolution. The effect on feedback is less clear but by increasing the mass resolution in simulations the escape velocity of the first halos is reduced. Since the gas temperature following a feedback event is not reduced (with increased resolution), heated gas will tend to orbit higher in the potential well and be more diffuse. It was noted that in the simulations with energy smoothing (even those with a high SFR) the formation of a dense central core was unavoidable. This might be partially associated with the delay between the first star formation and the first feedback. During this time it may be the case that gas can accumulate above the mass threshold at which it can be blown out. A probablistic star formation algorithm might improve this matter somewhat, although in the limit $N \rightarrow \infty$ these algorithms would be expected to coincide. However, it should be acknowledged that the dark matter halos in the sCDM picture have a strong central concentration and that the core accumulation may be a result of this.

As is widely known, core-halo angular momentum transport presents great problems for the sCDM picture. The main question mark is how bad the problem is for merging halos of unequal mass. It is unclear whether the limiting case of accreting a large number of low mass halos will lead to a system that has not lost a significant proportion of its angular momentum. Since the internal angular momentum of the final object is carried by the orbital angular momentum of the progenitors, there is reason to believe that it should be possible. However, higher resolution SPH studies (Thacker and Couchman, unpublished) indicate that it is almost impossible to avoid the formation of large objects from mergers of smaller ones without the angular momentum problem coming in to play – there are too many 'medium sized' halos that collapse within the galaxy halos. It is also possible to make insights without relying upon high resolution since the power spectrum relevant to galaxy formation is the approximately scale invariant $P(k) \propto k^{-2}$ (cooling times are shorter for the smaller halos, and the power spectrum is tilted more toward $P(k) \propto k^{-3}$). The simulations thus performed give a good indication of the properties of the low mass progenitors of galaxies – they too should suffer from an angular momentum deficit. However, since this is a problem for the internal angular momentum and not the orbital, the effect on the final object may not be a significant problem. In light of this argument, it is seen that smooth infall in simulations occurs as a result of a *lack of resolution*. Indeed the idea of smooth infall in sCDM is largely a misnomer (at least without some kind of feedback mechanism). In simulations, the minimum mass scale effectively keeps the gas supported against the collapse that would normally ensue given higher resolution. Thus the discs that are formed in SPH simulations are a result of this numerical resolution problem, and ideally detailed convergence studies should be undertaken. A number of authors have already hinted at this problem, but as yet no systematic attempt has been made to deal with it or its effects. This problem is examined in chapter 5.

The large variations in SFRs that may be produced by small changes in the parameter space remain a significant concern. The treatment of the high density end of the SPH solver is of particular importance, since it can strongly affect the SFR calculated, and further the treatment is not

performed in the same manner by all research groups. The development of a standard test case for dynamical star formation algorithms is necessary to facilitate the comparison of different research results. Unfortunately, it is far from clear what kind of a test case should be adopted. Simple rotating cloud collapse models are not adequate since they provide no representation of the hierarchical formation process or the effect of tidal fields. Also, since some algorithms are grid-based and some are particle-based, it may be necessary to adopt a suite of similar test cases.

McGaugh (1998) presents a survey of a number of reasons why sCDM is unable to form low surface brightness (LSB) galaxies similar to those observed. Nonetheless, it is still instructive to compare the SFRs calculated with those from deep observations of star forming galaxies. Studies of the global SFR, when adjusted for dust extinction, suggest that there is no decline between $z = 1$ to 4.5. Consequently, the self-gravity criterion imposed, causing an abrupt turn on of star formation at $z = 4$, does not fit the data. This should not be over-interpreted since it is a result of a lack of resolution, adding higher resolution would move the onset of star formation to progressively earlier times. Turning off the criterion allows star formation to proceed very early on, at $z = 8.3$, but it also allows star formation to occur in regions where the dark matter may still be dynamically dominant, which is undesirable. Using the continuum UV flux, the DEEP survey of the ***Hubble Deep Field*** derives SFRs from 0.14 $M_\odot$ yr^{-1} to 24.92 $M_\odot$ yr^{-1} for $q_0 = 0.05$ and $q_0 = 0.5$ values are 11 times lower (Lowenthal *et al.*, 1997). Note that these values are *not* corrected for dust extinction. At $z = 3$ the SFRs calculated (30-70 $M_\odot$ yr^{-1}) are higher than those found, although the selected sample are categorized as "large dwarf spheroidals" and, in contrast, at $z = 3$ the simulated galaxy already has a mass of 7.2×10^{10} $M_\odot$. More recently estimates of SFRs derived from Hβ emission, for a sample of $z = 3$ galaxies, have shown widely diverging values compared to the estimate from the UV flux (with both fluxes being uncorrected for dust extinction). The Hβ values are larger, by as much as a factor of 7, yielding SFRs in the range 20-270 $M_\odot$ yr^{-1} (Pettini *et al.*, 1998). It is difficult to see how the models presented can be tuned to produce SFRs in the region of 270 $M_\odot$ yr^{-1} since it would require an SFR normalization far beyond the conservative parameter space. High resolution would be necessary to derive values in this range whilst still maintaining a reasonable global SFR.

In summary, the accurate prediction of the morphology of galaxies requires high spatial and *mass* resolution. It has been shown that simple algorithms for calculating the SFR can produce reasonable results, provided that care is taken to explore the parameter space and understand the effects of changes in the algorithm. By introducing a parameter to control the lifetime of feedback events, it was shown that realistic morphologies can be produced in isolated disc galaxies.

Chapter 4

Parallel computation in simulation of galaxy formation

"640K ought to be enough for anybody."

–Bill Gates, 1981

4.1 Introduction

Because of the desire to make statistical predictions from simulations, it is vital to achieve as high a resolution as possible. High resolution ensures that statistical predictions are not affected by sample variance and hence that they are valid. The need to achieve high resolution separates simulation from, say, the numerical integration of a single variable partial differential equation. In this chapter a parallelization of the HYDRA simulation algorithm on shared-memory hardware is presented. The work presented allows simulations, with mass resolution an order of magnitude larger than those possible on workstations, to be conducted on mid-range symmetric multi-processor (SMP) machines. Detailed attention is paid to the architectural considerations of the computer hardware and also the structure of the serial code. A number of improvements to the serial code, which double the speed of execution, are also dicusssed.

The imperative of achieving high resolution has led to supercomputing having a well established position in numerical cosmology. This paragraph briefly reviews notable contributions to the development of the field. Seminal work on adapting particle-based methods, specifically treecodes, to vector architectures can be traced to work by Hernquist (Hernquist, 1990; Hernquist and Katz, 1989). This early work produced codes with execution speeds of order 100 MFlops, which at the time was almost a factor of 100 faster than the fastest implementation on a workstation. A vectorization of the P^3M grid code was undertaken by Summers (1993), following work performed in the plasma field (Nishiguchi *et al.*, 1985; Horowitz, 1987; Heron and Adam, 1989). This code also achieved a calculation speed of order 100 MFlops. Shortly after, both Ferrell and Bertschinger (1994; 1995) and Theuns (1994) adapted P^3M to the massively parallel architecture of the Connection Machine. This early work on massively parallel machines highlighted the need for careful examination of the parallelization process and the inherent difficulties in load balancing gravitational simulations. More recently Davé *et al.* (1997) have discussed the porting of the TREESPH code to parallel architectures using the Message Passing Interface (MPI) application program interface (API). Adaptive P^3M in combination with SPH has been effectively parallelized on the Cray T3D by Pearce & Couchman (1997) using Cray Adaptive Fortran (CRAFT). Brieu & Evrard (1998) discuss an implementation of P^3M for the IBM SP2. In an attempt to push the performance envelope of simulations, Macfarland *et al.* (1998) have developed an implementation of P^3M for the Cray T3E using the Cray shared memory (SHMEM) API. This code uses extensive static load balancing and by simulating the interaction of 1000^3 particles has performed the largest cosmological simulations to date (Evrard, 1999). Another noteworthy simulation is that of Warren *et al.* (1997) which used a parallel treecode to simulate the interaction of 3×10^8 particles on the ASCI Red supercomputer.

Supercomputing is required to simulate galaxy formation at high resolution. The primary reason for this is that doubling the linear resolution (achieved by increasing the particle number by eight) increases the computational task by a factor of approximately 20. The solution time per time-step increases by roughly a factor of 10 due to the increased number of particles and the Courant

condition asserts that the number of time-steps must be doubled because of the higher resolution. Thus the progression from a simulation with 64^3 particles to one with 256^3 particles results in a problem that is 400 times more computationally expensive. This is close to the performance scaling between workstations and massively parallel supercomputers.

The complexity of programming n-body codes on large scale supercomputers makes development inherently slow. Since most of these machines have distributed memory, message passing algorithms are necessary. However, the message passing paradigm is a highly complex programming model and codes that use global solution methods (Macfarland *et al.*, 1998; Brieu and Evrard, 1998), typically require man-years to write. Global addressing makes the task inherently easier, as can be seen in the implementation of the 'HYDRA' simulation code on the Cray T3D using CRAFT by Pearce and Couchman (1997). In view of this fact, the global addressing of the shared memory programming model renders parallelization a comparatively simple task. Additional motivation for development of a shared-memory code can be had from examining the computational scaling of the program. As indicated, doubling the linear resolution will lead to a task 20 times as computationally intensive. Thus shared memory SMP machines, which typically have 8-32 CPUs, provide an extremely important resource for problems that fall midway between workstations and massively parallel supercomputers. Also the design of these machines is becoming more and more commodity-based, which in turn reduces the cost significantly without decreasing performance. Because of the relatively low cost of these machines, large numbers of academics researchers to have access to the Giga-Flop class computing they provide.

Until recently, no portable API existed for shared memory programming. Hence programmers were reluctant to develop SMP codes since the resulting program was tied to a set of parallel instructions for a particular machine. On the other hand two portable APIs exist for message passing, namely Parallel Virtual Machine (PVM) and MPI. The advent of the OpenMP shared memory API (OpenMP Consortium, 1998) has remedied this situation. The standard has been adopted by all the major vendors and is a significant improvement on the old X3H5 draft ANSI proposal (Leasure, 1994). Hence as soon as compilers became available, the codes presented in this chapter were converted to OpenMP to render them as portable as possible.

It is interesting to note that clustered SMPs are rapidly becoming the architecture of choice for the 'next generation' of massively parallel supercomputers (*e.g.* ASCI Blue Mountain, Blue Pacific and Option White). Effective programming on these machines may well require an API combining the shared memory features of OpenMP and the message passing ability of MPI. Recent work on the "Treadmarks" library (Amza *et al.*, 1997) has attempted to bridge this gap. Treadmarks is a distributed shared memory (DSM) library designed to provide a shared memory programming model on distributed memory systems (*i.e.* PC clusters). The design of the system is based loosely around the "page faulting" method whereby pages of memory are passed around the distributed nodes as they are required. Sophisticated caching protocols are used to prevent this becoming inefficient. The recent work (Cox *et al.*, 1999) on Treadmarks has focused on incorporating p-threads to implement intra-node parallelism and combining this with the original message passing implementation for inter-node communication. These developments are interesting since they suggest that the shared memory programming model, supplemented by skillful programming, may be scalable beyond the traditionally perceived performance maximum of a few tens of processors.

Parallel algorithms are more difficult to program than serial ones. Most of the difficulty stems from the fact that a serial algorithm can be designed under the assumption that a consistent temporal order for the operations exists. For example in a loop over an index that runs from 1 to 100, it can be explicitly assumed that iteration 5, for example, follows iteration 4. In a parallel implementation of the code (strictly speaking 'loop level parallelism') the iterations of the loop are distributed among processors. Hence, many of the iterations may complete at the same time. Thus a code which relies on a particular temporal order of the iterations for correct execution will fail. A code which exhibits a dependence upon the ordering of loop iterations is said to exhibit a 'race condition'. Sometimes a race condition can be avoided by the use of 'locks' to prevent access to a variable which involves a race condition, other times additional copies of the variable can be spawned for each processor. More often than not, avoiding race conditions without suffering significant performance degradation involves a rewrite.

The organization of this chapter is as follows: First a short background review of computer architectures is provided to explain some of the motivation behind algorithm development. This is followed by an overview of the serial algorithm and its optimization. Next a detailed decomposition of the parallelization approach is given. Having described the methods, a review of performance and scaling is presented. The chapter is concluded with a brief review.

4.2 CPU architectures and parallel programming paradigms

Code performance on modern computers can be drastically affected by the architecture of the machine. Hence in this section, the architecture of Reduced Instruction Set Computer (RISC) processors is reviewed and attention is paid to how code must be optimized for them. The issues discussed are relative to any code.

4.2.1 Reduced Instruction Set Computer processors

The architecture of RISC CPUs incorporates a memory hierarchy with widely differing levels of performance. Consequently, the efficiency of a code running on a RISC processor is dictated almost entirely by the ratio of the time spent in memory accesses to the time spent performing computation. This fact can lead to enormous differences in code performance.

The first level of the memory hierarchy is the on-chip, 'level 1', cache memory. Fetching a memory element from the level 1 cache requires 1-2 CPU clock cycles and is typically 80-100 times faster than a fetch from main memory. Space constraints on the chip die usually mean that the level 1 cache is small, with 100 kb being the upper limit. The next level in the hierarchy is the level 2 cache which usually sits very close to the CPU and has an access time about 4-5 times slower than the level 1 cache, which is still 20 times faster than main memory. Larger level 2 caches are possible because there are no space constraints, however the cost of high speed memory rapidly becomes prohibitive. The largest current level 2 caches are of order 8 Mb. The final level in the hierarchy of solid state memory (as opposed to disk-based 'swap space') is the main memory for which the fetch time is of order 100 ns. Variants on the hierarchy exist, for example some SMP machines and the Samsung/Compaq Alpha CPU implement a level 3 cache, while the Hewlett-Packard PA-8500 CPU has a 1.5 Mb level 1 cache and no level 2.

When an access to main memory is made, a 'cache line' is retrieved which usually consists of the data element requested plus the next three words from the array. Once retrieved, the elements are placed into the (fast) level 1 cache. The length of a cache line is typically four words, although some machines can be tuned to use longer lines. If the algorithm is programmed such that the three additional elements retrieved are required very soon then this is an efficient process — one fetch has been required instead of four. Programs that exhibit this behaviour are said to exhibit good *cache reuse*, and can work very fast on RISC CPUs. Essentially an algorithm that accesses memory contiguously, and then performs a large number of operations on the retrieved data, will run fast. If only a small number of operations are performed on the retrieved data (as in a matrix multiplication), then the calculation becomes limited by the amount of data that can be moved from memory per second, which is commonly termed the *memory bandwidth*.

A second factor must also be considered: even if a program exhibits good cache reuse, it may still have a memory access pattern that is highly noncontiguous. Although programs like this are comparatively rare, they help to elucidate a subtle distinction in terminology and also illustrate the importance of memory access order. Suppose a program randomly selects elements from an array, and then performs a calculation based on the retrieved value plus the four preceding ones. The program exhibits good cache reuse, since the previous four elements are used again, however the random nature of the first element means that the code exhibits a number of *cache misses*. A cache miss occurs whenever the required data element is not held in the local cache memories. For a small enough array, all the elements can be stored in the cache memories. However, since the fastest cache memories are small compared to contemporary 3-dimensional grids, 100 kb at most, this situation is rare. It should be clear that a low number of cache misses is a direct consequence of good cache reuse, but a high number of cache misses does not have to imply bad cache reuse.

In summary, to exhibit good performance on a RISC processor, a code must exhibit both good cache reuse and a low number of cache misses. In practice, keeping cache misses to a minimum is the first objective since cache reuse is comparatively easy to achieve given a sensible ordering of the calculation (such as a FORTRAN loop) and often programs do not require significant cache reuse anyway (matrix multiplication for example).

4.2.2 Architecture of the SGI Origin 2000 supercomputer

The Origin 2000 server is currently the most popular mid- to large-scale SMP server. Many academic institutions have purchased these machines because of their flexibility and high performance. Although marketed under the pretense of being a shared memory machine, the Origin 2000 architecture is strictly 'distributed shared memory'. The memory of the system actually resides locally on each CPU board—not on a shared bus—and is transported from one CPU to another via a hypercube topology network. It is the hypercube topology that allows the Origin to scale up to 128 nodes, since as more nodes are added to the system the distance a memory word must travel from one CPU to another only increases with the logarithm of the number of CPUs. However, this

sophisticated architecture leads to memory access times that may vary quite significantly (from 100 to 1000 ns), which is known by the acronym NUMA for *Non-Uniform Memory Access*. To assure a consistent picture of memory across the system, a complex 'cache coherency' protocol is employed which carefully tracks where any memory element resides.

Since the Origin inherits a distributed memory architecture, efficient parallelization on it requires careful thought about data distribution and the way calculation is spread among processors. Note that if a code exhibits good cache reuse and a low number of cache misses, then the NUMA architecture should not impact code performance significantly since the memory overhead is removed. Conversely, codes with non-uniform memory access can be impacted quite severely. A worst case scenario can be imagined when all of the data resides on one processor node while the computation is distributed across every node. In this case a large bottleneck will occur at the node where the memory resides.

Compiler directives from SGI allow the data distribution to be controlled effectively. Both block and cyclic data distributions across memory nodes (plus combinations of the two) can be used to distribute data efficiently across the machine. A limit to the granularity of the distribution is provided by the minimum size of the memory pages, which is currently set by SGI to be 16kb.

4.3 Detailed breakdown and optimization of the serial algorithm

4.3.1 Review of the serial algorithm

Recap of particle simulation methods

When attempting to solve the gravitohydrodynamic equation set (equations 1.7- 1.11), for a large number of particles, algorithm efficiency is of vital importance. Solution of the equations of motion for gravity can be found using a pair-wise approach (see section 1.6.1), but this is extremely slow for a large number of particles (N greater than 10^4), since the calculation is of order N^2. An alternative approach is to use the Particle-Mesh method (see section 1.6.2), which uses a field representation for the density of the particle data. In converting to a field representation on a mesh, it is necessary to conduct an operation of order N. Once performed, a rapid solution to Poisson's equation may be found using a fast Fourier transform (FFT) which is order L^3 log L, where L is the size of the Fourier mesh. The total execution time using this method scales as $\alpha N + \beta L^3 \log L$, where α and β are constants.

The shortcoming of PM is that it is unable to resolve below a length scale set by the resolution of the mesh which is Fourier transformed (corresponding to the Nyquist frequency of the mesh). It is, however, possible to supplement this PM force with an additional short range force calculated from local particles on a pair-wise basis, yielding a force which is accurate below the resolution of the mesh. This is the P^3M method discussed in section 1.6.3. The execution time scales in proportion to $\alpha N + \beta L^3 \log L + \gamma \sum N_{pp}^2$, where γ is a constant and N_{pp}^2 corresponds to the number of particles in the short range force calculation within a specified region. The summation is performed over all the PP regions, which are identified using a chaining mesh. See Figure 4.1 for an illustration of the chaining mesh overlaid on the potential mesh. P^3M suffers the drawback that under heavy gravitational clustering the short range sum used to supplement the PM force slows the calculation down dramatically - the N_{pp}^2 term dominates as too many particles contribute to the short range sum. Although acutely dependent upon the particle number and relative clustering in a simulation, the algorithm may slow down by a factor between 10-100 or possibly more. While it is possible to resort to finer and finer meshes, memory limitations make this rapidly prohibitive and, additionally, computation becomes wasted on areas that do not require the higher resolution.

Adaptive P^3M (AP^3M, Couchman 1991), solves this problem by isolating regions where the N_{pp}^2 term dominates and solving for the short range force in these regions using FFT methods supplemented by a smaller number of short range calculations. This process is a repeat of the P^3M algorithm except that the FFT is now calculated with isolated boundary conditions. At the expense of a little additional bookkeeping, this method circumvents the slow-down dramatically. Dependent upon clustering, AP^3M is as much as 10 times, or more, faster than P^3M.

When implemented in an adaptive form, SPH is an order N scheme and fits well within the P^3M method since the short-range force supplement for the mesh force must coincidentally search to find the particles which are used in the SPH calculation. Hence, once a list of particle neighbors has been found, it is simple to sort through this and establish which particles are to be considered for the gravitational calculation and the SPH calculation. To incorporate SPH into AP^3M it is

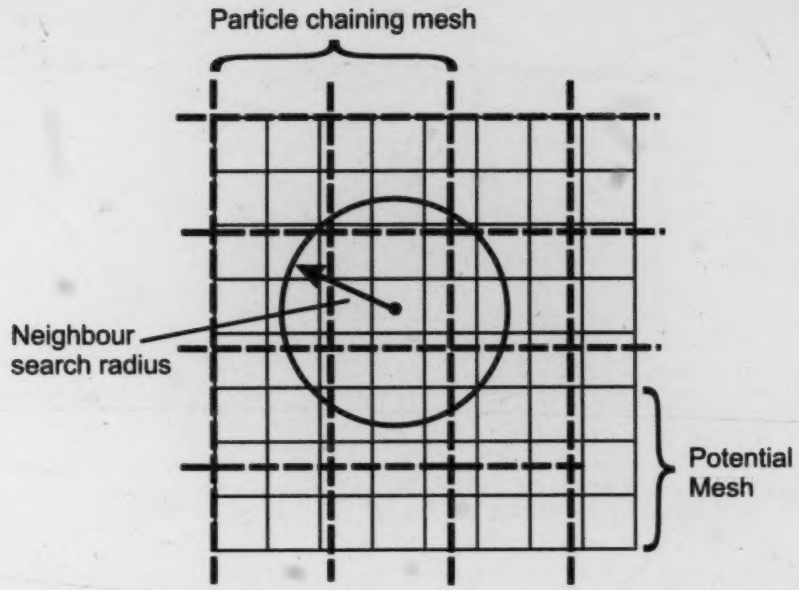

Figure 4.1: Overlay of the chaining mesh on top of the potential mesh to show spacing and the search radius of the short range force.

only necessary to make coordinate scalings and do minor book keeping. The combined adaptive P^3M-SPH code, 'HYDRA', in serial FORTRAN 77 form is available on the World Wide Web from http://coho.astro.uwo.ca/pub/hydra_consort/hydra.html.

The solution cycle of one time-step may be summarized as follows,

1. Assign mass to the Fourier mesh.
2. Convolve with the Green's function using the FFT method to get potential. Difference this to recover mesh forces in each dimension.
3. Apply mesh force and accelerate particles.
4. Decide where it is more computationally efficient to solve via the further use of Fourier methods as opposed to short-range forces and, if so, place a new sub-mesh there.
5. Accumulate the gas forces (and state changes) as well as the short range gravity for all positions not in sub-meshes.
6. Repeat 1-5 on the sub-meshes until forces on all particles in simulation have been accumulated.
7. Update time-step and repeat

Note that the procedure of placing meshes is hierarchical in that a further sub-mesh may be placed inside a sub-mesh. This procedure can continue to an arbitrary depth but, typically, speed-up only occurs to a depth of six levels of refinement.

Review of the serial algorithm

In this section, an overview of the main subroutines is given to provide reference for the later discussion on parallelization of the code. The necessary input files for the program are a parameter file `prun.dat` and a data file (containing particle positions, velocities and hydrodynamic quantities). The call tree of the FORTRAN 77 implementation is as follows (subroutine names are capitalized):

```
MAIN:
   STARTUP: read in parameters and data
```

```
INUNIT: define the units
loop until finished:
UPDATERV: Predict-Evaluate-Correct step
|    ACCEL: acceleration including hubble drag; timestep evaluation
|        FORCE: acceleration evaluation
|           RFINIT: initialize refinements
|           REFFORCE: see below
|           CLIST: create particle lists for refinements
|           loop over refinements:
|           |  LOAD: load particles into refinement
|           |  REFFORCE: see below
|           |  ULOAD: unload particles from refinements
|           end loop
|     INFOUT: write summary file
|  OUTPUT: write out data and backup files
end loop
end

REFFORCE:

 GREEN     : evaluate, or read, green's function
 MESH      : evaluate PM accelerations
 LIST      : sort particles into search cells
 REFINE    : determine the position of sub-refinements
 SHFORCE   : tabulate PP force and potential
 SHGRAVSPH : apply PP and SPH forces;
             write out diagnostic information
```

This call tree is given in diagrammatic form in Figure 4.2. The following paragraphs review all the important subroutines within the code. A brief mention is given to the work in each routine and the programming constructs used. The terms 'gas particle' and 'SPH particle' are used interchangeably.

startup Opens the file `prun.dat`, which has the following format:

```
filename            !name of input data file
irun                !4 digit number of run
istop,itdump        !iteration to stop at, iterations between data back-up
dtnorm              !the time step normalization factor (<1)
sft100              !the desired effective Plummer softening in Mpc / h
nlmx                !number of refinement levels
```

The data file containing all the particle data has the following FORTRAN unformatted structure,

```
ibuf,ibuf1,ibuf2
rm[N]
r[3,N]
v[3,N]
h[N]
e[N]
itype[N]
dn[N]
```

`ibuf,ibuf1,ibuf2` contain input parameters for the gravity and sph-solvers (such as the density parameter, size of simulation box). A full summary of the parameters and formatting of the data is given in the HYDRA documentation (Couchman *et al.*, 1996). The array `rm` of size N, gives the mass of each particle (for isolated simulations a value of 1 is 10^{10} $M_{\odot}$, while for comoving simulations mass units are scaled relative to the density parameter for the system, in such cases only the mass ratio between particles becomes important). `r` contains the particle position data and is thus of size 3N. Similarly `v` contains the velocity data, `h` the smoothing lengths for SPH particles, `e` the temperature for SPH particles, `itype` the particle type (gas, star or dark matter) and finally `dn` contains the density values for SPH particles. For star and dark matter particles the hydrodynamic quantities are set to zero.

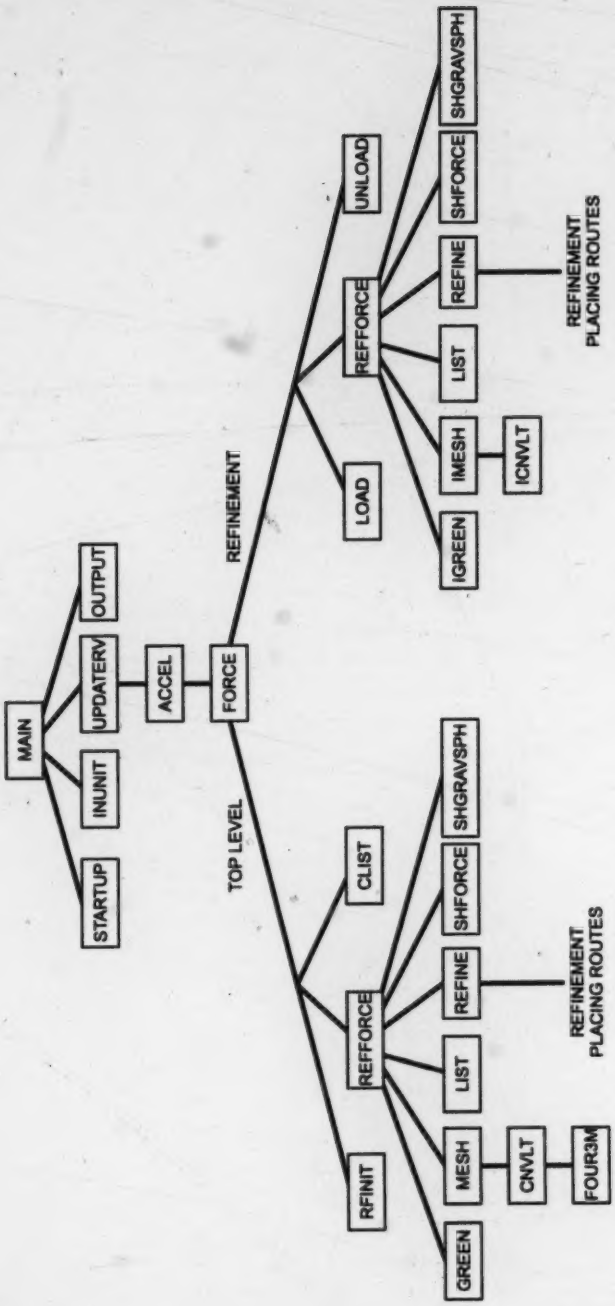

Figure 4.2: Call tree of the serial HYDRA algorithm. Only significant subroutines are shown for clarity. The refinement routines are the same modulo the effect of periodic wrap-around in the top level.

inunit The units used in the simulation are calculated from the values loaded in from the start-up files. A set mass unit is applied for fixed box-size simulations (10^{10} $M_{\odot}$) while for comoving simulations the value scales with box size and particle number. As previously mentioned, for periodic simulations the actual numbers in the `rm` array represent a ratio between particle masses, the total mass being scaled relative to the density parameter. Length units are calculated from the size of the simulation box supplied in the buffers in the particle data file. Time units are set at 1×10^{10} yr for isolated simulations while for comoving simulations they are scaled relative to age universe calculated for the given cosmological parameters (again supplied in the particle data file). Velocity, density and energy (temperature) units are then derived from the length, mass and time units. The bulk of the work in this routine is a loop over all particles to calculate the total mass.

updaterv After the main subroutine begins a time-step loop, **updaterv** is called each time to update the particle data. A Predict-Evaluate-Correct integrator is used where the positions are first predicted forward, and then a correction is applied, which is calculated using the particle data at the predicted positions (see CTP95 for a full discussion). After predicting the particle data forward using a loop over particles, **updaterv** makes a call to **accel** to evaluate the accelerations. The correction step is then applied using the values from the acceleration array array found in **accel** . This step is once again performed using a loop over particles. The last part of the subroutine makes a call to one of the output routines to dump diagnostics for the time-step.

accel Both the acceleration data and energy change for each particle are initialized to zero (a loop over all particles). After a call to **force** the time-step criterion is assessed by sorting through the acceleration and particle data (another loop over all particles). In comoving simulations, this limit is then compared to that from the Hubble expansion. If the simulation cube is a comoving box, then a correction is applied to the acceleration and energy change arrays to account for the comoving coordinates used in **force** and **shgravsph** (see appendix A). Once again a loop over the particles is used.

force Control of the adaptive algorithm begins in this subroutine. It first sets up a work area for the gravitational routines to avoid excessive memory use and then calls **rfinit**, primarily to initialize the refinement (sub-grid) data arrays, such as position and grid size, used in the adaptive scheme. Following this, the top level grid of the hierarchy is evaluated by a call to **refforce**. A call to **clist** then rearranges the linked list structure calculated in **refforce** so that the first elements of the list carry the indices of particles which are to be loaded into refinements. Following this a loop over the refinements, with an `IF...GOTO` break clause, evaluates the refinements. Within the loop each iteration must make successive calls to **load** (to load refinement data into arrays), **refforce** (to evaluate the refinement) and **uload** (to unload the data from the refinement). Clearly the call structure for refinements is closely similar to that of the top level grid and the main evaluation routine **refforce** is used repeatedly. This similar structure is evident in the call tree in Figure 4.2.

rfinit Data arrays which store the refinement data (such as grid size, position of the left-hand corner and number of particles in the refinement) are initialized. The size of the chaining cells necessary to maintain the desired force error is also calculated. There is almost no work in this routine in comparison to others.

clist During the call to **refforce** a linked list structure is calculated (the **list** subroutine) and the refinements are placed. To ease bookkeeping in the refinement section of the code, the linked list structure is rearranged so all the particles which are loaded in a refinement can be found simply by inserting into the list at the correct point. This calculation involves a loop over all particles, but is somewhat complicated in that the insert point into the list for refinement 1,2,... must be stored in an auxiliary array.

load Refinement data arrays are constructed, with indices running from `1..m`. Loading of the particle data is performed using a loop over the new index. The indices of the particle data required to form the refinement data array are stored in the reordered linked list (*i.e.* the list structure after it has been rearranged in **clist** or **uload**). Thus the loading consists of statements, `rm(j)=rm(i)`, where `j`$= 1 \ldots m$ and `i` is found from the linked list.

uload While **load** is responsible for setting up data for a refinement, **uload** inserts the values found during the evaluation of the refinement into the global particle arrays. The loop structure is the same **load**, but an additional rearrangement is done on the linked list structure so that it contains

the list of particles which are to be loaded into any refinements placed. This step is analogous to the role of **clist** for the top level grid.

refforce This routine controls the calling of the subroutines performing the bulk of the computational load. Firstly, depending upon whether the refinement being calculated has isolated or periodic boundary conditions, the green function for the Fourier transform is calculated (by calls to **green** or **igreen**). Following this, either **mesh** or **imesh** is called to evaluate gravitational force from the grid. Next **list** is called to calculate the linked list to be used in the short range force calculation. To place refinements over data which is too clustered (which must be done before the call to the short range force routine) **refine** is called. Finally **shgravsph** is called to evaluate the short-range gravity and hydrodynamic forces.

mesh & imesh To calculate the grid-based gravitational force, the particle mass data is first assigned to a density grid using the triangular shaped cloud (TSC) assignment function (Hockney and Eastwood, 1988). This step is a loop over particles. Next, a call to the convolution routine **cnvl** (or **icnvlt**) performs the Fourier convolution with the green function and thus the potential array is recovered. The potential array is then differenced, using nested loops, to construct the force array. Finally, a loop over particles allows the accelerations to be calculated using the same interpolation operation as the mass assignment step at the beginning of the routine.

cnvl & icnvlt These routines are responsible for performing the Fourier convolution of the density array with the green function to recover the potential. For the periodic routine, **cnvl**, the Fourier transform is performed by a call to the routine **four3m**. Next, the convolution in Fourier space is conducted using three nested loops over the x,y and z directions during which the potential energy is also calculated. Once the convolution with the green function is complete, the inverse Fourier transform is performed by another call to **four3m**. For the isolated routine, the convolution is more sophisticated since the calculation is optimized to avoid unnecessary calculation over regions with zero padding. First lines parallel to z are Fourier transformed (utilizing auxiliary arrays), and then the convolution over planes is performed two sheets at a time. As in the periodic calculation, the potential energy of the grid is calculated during the convolution. Once the plane convolutions have been performed, the inverse transform of lines parallel to z is performed to recover the potential array.

four3m This routine is a wrapper for the Numerical Recipes 3-d Fourier transform **fourn**. For large 3-d FFT's the **fourn** routine is inefficient due to the stride-length exceeding the cache size of the CPU. **four3m** solves this problem by rewriting the FFT as a sequence of FFT's over sequential lines (or planes) and using **fourn** to conduct these FFT's. The routine is typically 5 or 6 times faster than **fourn** for large problem sizes.

list The linked list, used in the short range gravitational and hydrodynamic force calculation, is calculated in this subroutine. Since the simulation cube is divided up into chaining cells, the size of which is set by the local neighbour search radius, the linked list is actually a composite of the linked lists for all the cells. The list is built by a simple insertion process and is ordered in the z-direction to improve performance in the short range calculation. During construction of the linked list arrays are constructed that contain the number of particles in each chaining cell (`nhc`) and also the index of the first particle in each cell (`ihc`). The second array is necessary to insert into the correct entry point within the linked list.

refine Placing refinements is a very complicated optimization problem, and consequently, only a brief discussion of this routine is given. Firstly, a search is performed over the `nhc` array to identify high density regions which need to be removed from the current refinement (so that the algorithm does not become inefficient). A first estimate of the optimal grid size is made for each of these regions. Then a sequence of iterations checks for overlaps and whether merging two regions would be optimal. Once all overlaps have been identified and removed, and the refinement sizes have been iterated to an optimal configuration the refinement data (such as coordinates, number of particles and Fourier transform size) are stored.

shgravsph Far and away the largest part of the calculation is spent in this routine which calculates the short-range gravitational forces and hydrodynamic evolution. A nest of 3 loops in the x,y, and z directions is used to sort through the chaining cells. The interactions for the particles in each cell are calculated by sorting through the linked lists for the neighbouring cells and identifying whether the particles found fall within the maximum neighbour search distance. Note that during the search,

it is checked to see whether particles have been placed in a refinement. If so, the algorithm quickly sorts through the particles to increment particle counters but does not calculate interactions (these will be done in the refinement). A caveat to this procedure is that since the SPH search length may be larger than the gravitational search radius, it is sometimes necessary to evaluate SPH particles in the current refinement. In this case, only the SPH forces are calculated and the particle type is switched to "done". The gravitational forces are then calculated in the refinement. This can become a significant inefficiency.

4.3.2 Changes to artificial viscosity, equations of motion and energy

All the changes made to the code (as detailed in CTP95) are motivated by the findings in chapter 2. A detailed examination of the differences between SPH implementations may be found therein. Although this section contains a partial repeat of material in chapter 2 it is included to provide a quick reference against the code discussed in CTP95 and the later HYDRA v2.0 production code.

The artificial viscosity has been changed to a pair-wise version. The summation of P/ρ^2 terms in the equation of motion and energy is extended to include a term Π_{ij}, which is given by,

$$\Pi_{ij} = \frac{-\alpha\mu_{ij}\bar{c}_{ij} + \beta\mu_{ij}^2}{\bar{\rho}_{ij}} f_i, \tag{4.1}$$

where,

$$\mu_{ij} = \begin{cases} \bar{h}_{ij}\mathbf{v}_{ij}.\mathbf{r}_{ij}/(r_{ij}^2 + \nu^2), & \mathbf{v}_{ij}.\mathbf{r}_{ij} < 0; \\ 0, & \mathbf{v}_{ij}.\mathbf{r}_{ij} \geq 0, \end{cases} \tag{4.2}$$

$$\bar{\rho}_{ij} = \rho_i(1 + (h_i/h_j)^3)/2, \tag{4.3}$$

and

$$f_i = \frac{|<\nabla.\mathbf{v}>_i|}{|<\nabla.\mathbf{v}>_i| + |<\nabla\times\mathbf{v}>_i| + 0.0001c_i/h_i}. \tag{4.4}$$

with bars being used to indicate averages over the i, j indices. Shear-correction (Balsara, 1995; Navarro and Steinmetz, 1997), is achieved by including the f_i term which reduces the artificial viscosity in shearing flows (see chapter 2).

The equation of motion is the same as previous versions, modified to include the new artificial viscosity and kernel averaging:

$$\frac{d\mathbf{v}_i}{dt} = -\sum_{j=1, r_{ij}<2h_i}^{N} m_j \left(\frac{P_i}{\rho_i^2} + \frac{\Pi_{ij}}{2}\right) \nabla_i \overline{W}(\mathbf{r}_i - \mathbf{r}_j, h_i, h_j)$$

$$+ \sum_{j=1, r_{ij}<2h_j}^{N} m_j \left(\frac{P_j}{\rho_j^2} + \frac{\Pi_{ji}}{2}\right) \nabla_j \overline{W}(\mathbf{r}_i - \mathbf{r}_j, h_j, h_i). \tag{4.5}$$

where,

$$\overline{W}(\mathbf{r}_i - \mathbf{r}_j, h_j, h_i) = [W(\mathbf{r}_i - \mathbf{r}_j, h_i) + W(\mathbf{r}_i - \mathbf{r}_j, h_j)]/2 \tag{4.6}$$

Similarly the energy equation is the same as that of previous versions, modified to include the new artificial viscosity:

$$\frac{d\epsilon_i}{dt} = \sum_{j=1, r_{ij}<2h_i}^{N} m_j \left(\frac{P_i}{\rho_i^2} + \frac{\Pi_{ij}}{2}\right) (\mathbf{v}_i - \mathbf{v}_j).\nabla_i \overline{W}(\mathbf{r}_i - \mathbf{r}_j, h_i, h_j). \tag{4.7}$$

4.3.3 Revised time-step criteria

To find the value of the time-step, dt, the particle lists are searched to establish the time-step limitations of the acceleration, dt_a, and velocity arrays, dt_v (CTP95). The revised code introduces a further time-step criterion, dt_h, which prevents particles traveling too far within their smoothing radius,

$$dt_h = \min_i \left(\frac{h_i}{|\mathbf{v}_i^r|}\right), \tag{4.8}$$

where $|\mathbf{v}_i^r|$ is the maximum relative velocity between particle i and its neighbour particles (see chapter 2). dt is then calculated from

$$dt = \kappa \min(0.4dt_v, 0.25dt_a, 0.2dt_h), \tag{4.9}$$

where κ is a normalization constant that is taken equal to unity in adiabatic simulations. In simulations with cooling the largest density contrasts that develop sometimes require $\kappa = 0.5$.

Note an exact consideration of the Courant condition,

$$dt < \min_i \left(\frac{h_i}{c_i} \right), \tag{4.10}$$

where c_i is the sound speed for particle i, is not made, primarily because in an SPH code particle velocities are directly responsible for the propagation of hydrodynamics. For example a longitudinal pressure wave is transmitted by the particles oscillating about their equilibrium position. Of course the h/v criterion is not a direct proxy for the Courant condition, although it should be a good approximation. Further, in CTP95 it was found that accuracy requirements from the acceleration time-step constraint tended to more stringent than the Courant condition.

4.3.4 New smoothing length update algorithm

The smoothing length update algorithm has been revised to reduce the fluctuation in neighbour counts noted in previous versions. The procedure for updating h is as follows:

- A weighted neighbour count is made that weights near neighbours more than far. The precise weighting function is,

$$W_{nn}(r/h) = \begin{cases} 1, & 0 \le r/h < 3/2; \\ \pi W_s(4(r/h - 3/2)), & 3/2 \le r/h \le 2, \end{cases} \tag{4.11}$$

  where W_s is from equation 2.4.

- A prediction for the correction of h, s, is calculated from $s = (N_s/N_i^{n-1})^{1/3}$ where N_s is the desired number of neighbours (52) and N_i is the weighted neighbour count.

- The value of h at the next time-step, n, is then calculated from,

$$h_i^n = h_i^{n-1}(1 - a + as), \tag{4.12}$$

  where,

$$a = \begin{cases} 0.2(1 + s^2), & s < 1; \\ 0.2(1 + 1/s^3), & s \ge 1. \end{cases} \tag{4.13}$$

This new algorithm has been tested extensively and has been found to offer numerous advantages over the previous implementation (see chapter 1, for a review of the effect of each step).

4.3.5 Test-bed simulation

During development of the code, a $2{\times}64^3$ simulation, known as the 'Santa Barbara' cluster simulation was used. This simulation has been used extensively to compare the performance of a number of different hydrodynamic cosmological structure formation codes (Frenk *et al.*, 1999). The simulation follows the evolution of a galaxy cluster with a total mass of 1.1×10^{15} $M_\odot$ at $z = 0$, in a box size of 64 Mpc. No allowance is made for gas cooling and the mass resolution in the simulation (6.0×10^{11} for dark matter, 6.7×10^{10} for gas) is insufficient to determine individual galaxies. However, the simulation does resolve the structure of the gas lying in the dark matter potential of the cluster, allowing profiles of temperature, density and X-ray luminosity to be constructed.

The findings of the paper, in relation to the differences among codes, were complicated by the fact that different simulations used different particle numbers. Also there was a minor synchronization error across simulations due to some authors choosing to start the simulation earlier than others. Nonetheless there were a number of useful conclusions drawn that indicate the variance to be expected among codes:

1. The dark matter structure was similar across all simulations, with the variations in the radial density profile being bounded by ±20%.

2. The SPH simulations predict a lower value for the central entropy than the adaptive mesh refinement and finite-difference codes.

3. Radial temperature profiles varied by a similar amount (±20%) as did the gas density profile. All simulations showed a flattening of the gas profile in the inner core relative to the dark matter profile.

4. X-ray luminosity showed the largest scatter since it is dependent upon the density squared. Variation at the ±100% level was observed in the radial emission profile. This suggests a correct calculation of the luminosity profile actually requires very high resolution.

Given the changes to the HYDRA algorithm detailed in this thesis compared to v1.4, which was used to run the data presented in the Frenk *et al.* (1999) paper, it is useful to describe the changes in the solution found. This should highlight any noteworthy changes, such as a change in the entropy value.

The simulation was rerun with the revised code and it required significantly more steps than the previous version (2919 versus 1826). This is a consequence of two factors. Firstly, the new code has $h_{min} = \epsilon_{S2}/2 \simeq \epsilon_{Plummer}$ which allows higher densities to be resolved than version 1.4 which had double the value for h_{min}. Consequently larger accelerations are present, resulting in shorter time-steps. Secondly the dt_h time-step criterion is more stringent than that in version 1.4, and leads to more time-steps. The force error in the new run was also lowered to 5% (maximum) compared to the 7.7% chosen in the old run.

Figure 4.3 displays plots of the gas density, temperature, entropy and X-ray emission. The residuals (defined as $(x_{old} - x_{new})/x_{new}$) vary in the range reported in the comparison paper. A detailed analysis of these results is not warranted, however, comparison of the results to the remainder of the SPH implementations studied in the paper shows the new code is more in line with these results than the old. In particular the peak of the X-ray emission is more central and the entropy values are lower.

4.3.6 Optimizing the memory access of the serial code

As indicated in the general overview at the beginning of this chapter, codes which achieve a low number of cache misses and high cache reuse run well on RISC processors. While this is not a guarantee of a fast algorithm, it is usually a good measure. For example, the N^2 particle-particle method of calculating gravitational interactions has a very high computation to memory access ratio. Since the HYDRA code spends a significant amount of time accessing arrays in the short-range force calculation, the method used to access the data, namely linked lists, should be examined thoroughly.

Linked lists (where the list array is denoted `ll`) are a common data structure used extensively in particle-in-cell type codes (see Hockney and Eastwood, 1988, for an extensive review of their use). They are comparatively easy to compute and allow simple insertion of elements into the chain: to insert an object, one simply changes the value of the linked list for the preceding object to link to the new one and the linked list value for the new object links to the old object.

For a list of particles which is cataloged according to cells in which they reside, it is necessary to store an additional array which holds the label of the first particle in the list for a particular cell. This array is denoted `ihc` for Integer Head of Chain. Thus to search for the particles in a particular cell, the cell index (given by x,y,z coordinates in 3 dimensions) is used to index into `ihc` and recover the first particle in the list, particle `i`. The second particle in the list is then given by `j=ll(i)`, the third particle is given by `k=ll(j)` and so on until the particle m for which `ll(m)=0` is found, indicating that there are no more particles in the cell. In FORTRAN this procedure is usually programmed using an `IF...THEN...GOTO` structure, with the loop exiting on the `IF` statement finding a value of zero in the linked list. Because of this implementation, compilers are unable to make significant optimizations of code within the loop. Since the loop 'index' (the particle index `i`) is found recursively the compiler cannot make decisions about a number of optimization processes, particularly software pipelining, for which loops are usually better. Aside from issues of compiler performance, *if the particles' indices are not ordered in relation to the z-axis then there will usually be a cache miss in finding the element* `ll(i)` *within the linked list array.* This is because the indices thus found, namely `i,j,...,m` are not guaranteed to be contiguous (ie they are not `i,i+1,...,i+n`). This has another consequence within the particle data arrays, *i.e.* the arrays storing positions and velocities. Since the particle indices found from the linked list are not contiguous, *a cache miss is likely to occur in the particle data too.* A number of arrays must be accessed to recover the particle data, and consequently the removal of the cache miss associated with the particle indices should improve performance significantly.

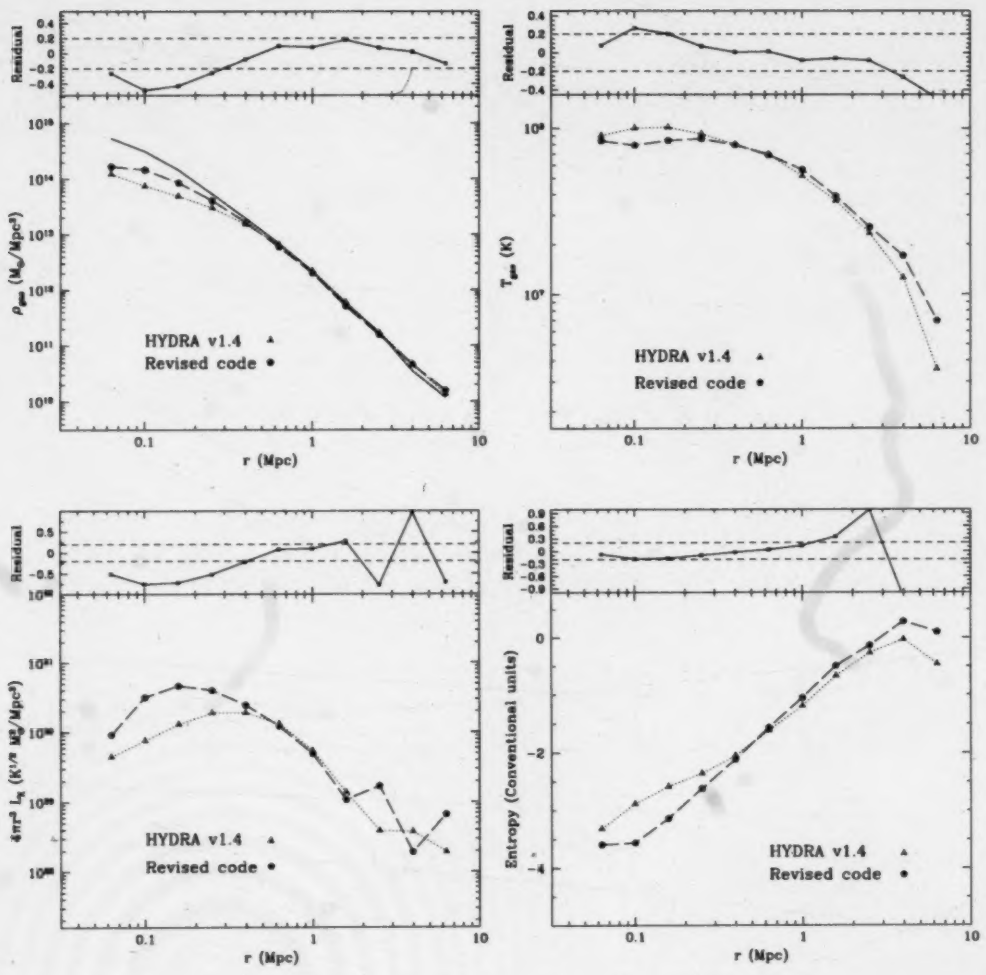

Figure 4.3: Comparison of the 'Santa Barbara' cluster simulation radial profiles produced by the updated SPH implementation compared to the old version. Starting clockwise from top left, the gas density, temperature, entropy and X-ray emission are shown. The gas density plot also includes the dark matter density profile scaled down by the mass ratio. The changes to the algorithm produce a code in broad agreement with the remainder of the SPH implementations in the paper. The outer lying values found in HYDRA v1.4 are brought closer in-line with the remaining codes.

The first step that may be taken to improve the situation is to remove the cache misses associated with the searching through the linked list. To do this the list must be formed so that it is *ordered*. In other words the first particle in cell `j`, is given by `ihc(j)`, the second particle is given by `ll(ihc(j))`, the third by `ll(ihc(j)+1)` *et cetera*. This ordered list also allows the short range force calculation to be programmed more elegantly since the `IF..THEN..GOTO` structure of the linked list can be replaced by a loop. However, since there remains no guarantee that the particle indices will be ordered, it remains difficult for the compiler to make some optimizations, although the situation is certainly far better than with the linked list. Note that the programming of this algorithm is comparatively simple since the same data structures are retained from the linked list code. Tests performed on this ordered list algorithm show that a 30% improvement in speed is gained over the linked list code (see Figure 4.4). The drawback remains that the particle indices are still not ordered, and so access to the position arrays, for example, will still exhibit cache misses.

As has been discussed, cache misses can be avoided in the particle data arrays if the list of particles is ordered such that for N particles in a cell the particle indices run from `j,..,j+N` where `j` is the index of the first particle in the cell. Note that the value of `j` is set by calculating the number of particles in previous cells. For example, the first particle in cell 2 has an index one more than that of the last particle in cell 1, similarly the first particle in cell 3 has a index one more than the last particle in cell 2, and so on. This idea is not new and has been discussed in the literature (Anderson and Shumaker, 1995; Decyk *et al.*, 1996; Thacker *et al.*, 1998; Macfarland *et al.*, 1998; for example). However, prior to the implementation presented here only the work in Macfarland *et al.* (1998) and Anderson and Shumaker (Anderson and Shumaker, 1995), actually revised the code to remove linked lists. Most other implementations of particle ordering do so every ten or so steps to keep the particles close to being ordered. If particle ordering is implemented implicitly within the code (*i.e.* the linked list is removed completely), then a permutation of the particle indices occurs from step to step. Furthermore, since the adaptive refinements use the same particle indexing method, the particle ordering must be done within the data loaded into a refinement, *i.e.* hierarchical rearrangement results from the use of refinements.

The step-to-step permutation is quite simple to calculate: first the particle indices are sorted according to their z-coordinate and then indices are simply changed to run from the calculated first particle index (*i.e.* one more than the last index in the previous cell) to this index plus the number of particles in the cell minus one. Consequently, if it is desired to keep track of particle indices in relation to the starting configuration, an additional array must be stored that is sorted using the same permutation which is performed on the particles. It is important to note that this method of particle bookkeeping removes the need for an index list of the particles (although in practice this storage is taken by the array which stores the permutation of indices). All that need be stored is the particle index corresponding to the first particle in the cell, *i.e.* the head of chain (`ihc`), and the number of particles in the cell (`nhc`). Thus for chaining cell, n, the loop is from the particle index `ihc(n)` to the particle index `ihc(n)+nhc(n)-1`. This method is known as *particle reordering* and is extremely efficient. There is no possibility of a cache miss in the particle index list since it has been removed completely. Secondly, the particle data is fetched contiguously resulting in a low number of cache misses and also good cache reuse. This is easily seen since when retrieving particle data for particle `i`, particles `i+1,i+2,i+3` will be used next. This algorithm is so efficient that the speed of the HYDRA simulation algorithm *more than doubled*. For example, at the end of the Santa Barbara cluster simulation, the execution time was reduced from 380 seconds to 160 seconds. A comparison plot of the performance of a linked list, ordered list and ordered particle code is shown in Figure 4.4.

4.4 Parallelization of HYDRA on shared memory multiprocessors

SMP Fortran Implementation

FORTRAN remains a highly popular language amongst scientists. Although primarily due to the large amount of legacy code, this popularity is also maintained by the continued high performance of FORTRAN compilers. Indeed, high performance is especially important in large N-body simulations where the total number of calculations borders on a Peta-Flop. A further reason for the popularity is the ongoing evolution of the instruction set to include modern programming paradigms such as dynamic allocation of memory in the FORTRAN 90 standard.

From the programming viewpoint if algorithmic development is an ongoing project (as opposed to having a predefined production code) then it is extremely useful to keep the program code as simple as possible. Fortunately the parallel extensions in the OpenMP programming model are comparatively simple, in turn making the maintenance of parallel codes trivial. Further, the shared memory environment allows global addressing which saves the trouble of implementing explicit message passing.

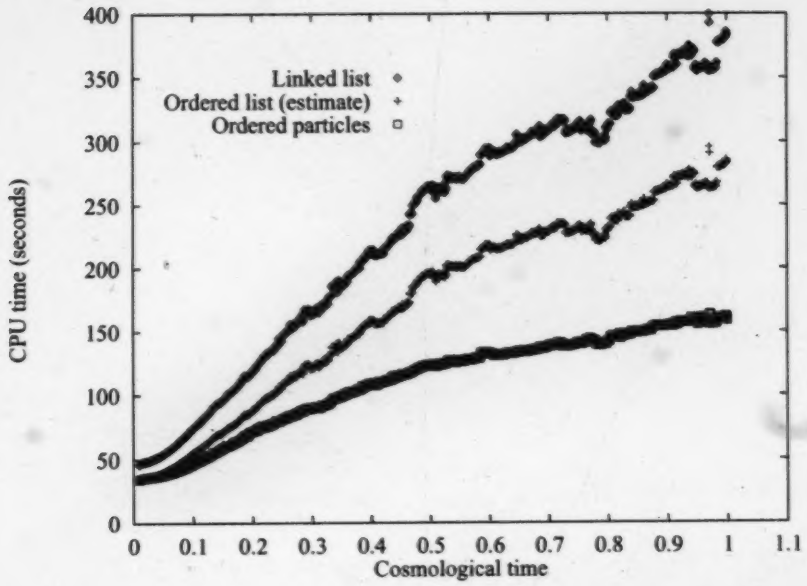

Figure 4.4: Execution time for the Santa Barbara cluster simulation for codes using different list structures. The linked list corresponds to the current publicly available code. The entire simulation was not run using the ordered list code, but instead the time for the final step was calculated and the linked list data was in turn scaled to match this value. At the end of the simulation the ordered particle code was 2.4 times faster than the linked list and 1.8 times faster than the ordered list.

The parallel code discussed, was first developed on an SGI Power Challenge and later on an Origin 2000 system using MIPSPro Fortran (Silicon Graphics Incorporated, 1997). During code development the OpenMP API became available and a switch to the MIPSPro 7.2.1 compiler which is OpenMP compliant, was made. The dynamic load scheduling option has been used extensively, and it has proven particularly useful for load balancing some difficult situations. The code was parallelized subroutine by subroutine so that any errors could be immediately tracked. Because shared memory programming only involves insertion of parallel pragmas it can be immediately tested against the serial code to assure correctness. This checking was performed at each stage of development, and some parts of the code (specifically the energy calculation) were rewritten at double precision so that round-off errors could be removed as a potential source of differences between the parallel and serial code.

The call tree for the parallel algorithm is presented in Figure 4.5. It is similar to the serial code in concept, but is made more complicated by the distinct division between larger refinements calculated across the whole machine, and small ones which are performed as a task farm (see section 4.4.10 for a discussion). A complete discussion of the parallelization approach, for each subroutine with a significant workload, is given in sections 4.4.3- 4.4.13.

In the discussion the term processor element (PE) is used to denote the number of parallel threads of execution. Since only one thread of execution is allotted per processor, this number is equivalent to the number of CPUs, and the two terms are used interchangeably.

4.4.1 Load balancing options provided by the OpenMP standard

Parallelization on SMP architectures is achieved using parallel do loops where the iterations of the loop are distributed amongst the processors. Load scheduling—assuring that one set of iterations does not take significantly longer than another set of the iterations—can be achieved in a number of different ways. The OpenMP directives allow for the following types of iteration scheduling,

- static scheduling - the iterations are divided into chunks (the size of which may be specified

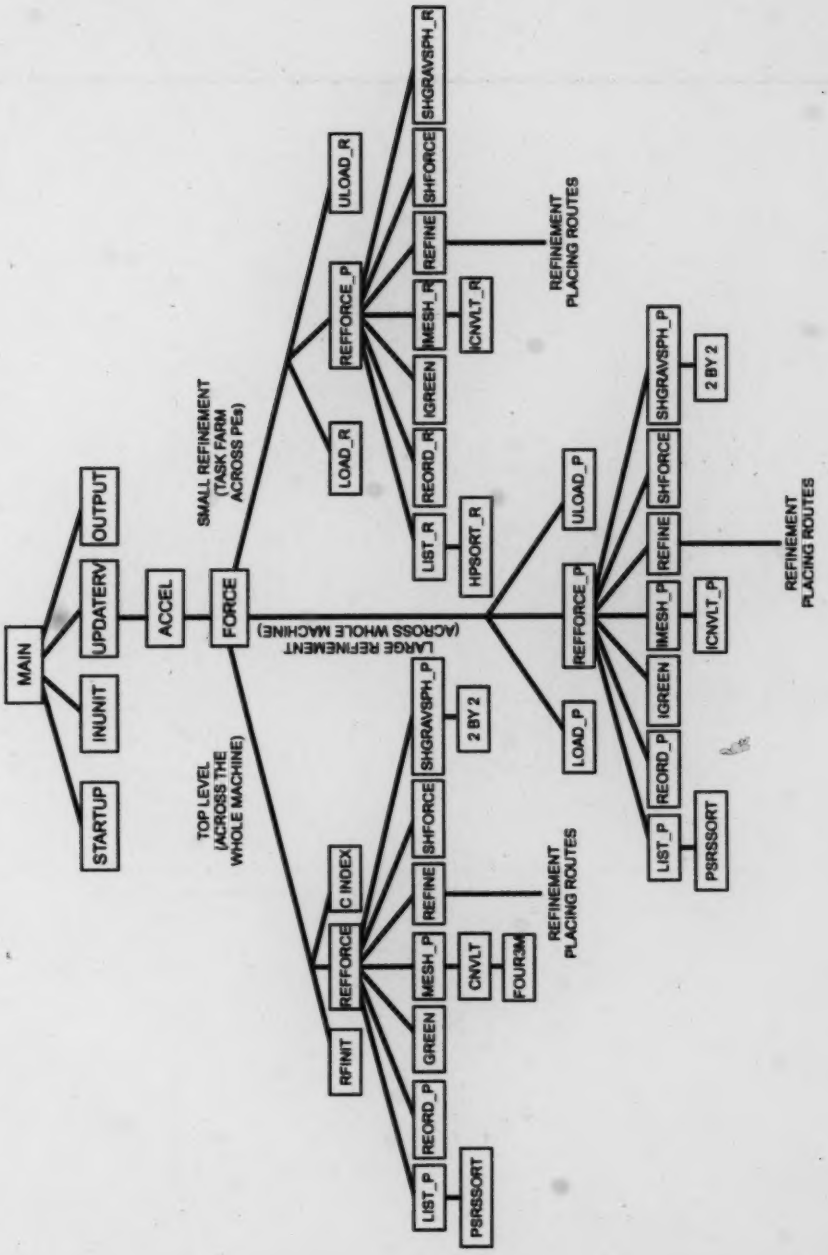

Figure 4.5: Call tree of the OpenMP based parallel HYDRA algorithm. Only significant subroutines are shown for clarity. The largest conceptual difference between the parallel code and the serial code is the treatment of large refinements across the whole machine.

if desired) and the chunks are distributed across the processor space in a contiguous fashion. A cyclic distribution can also be chosen so that (on N processors) a single processor receives iteration `i,i+N,i+2N,...,i+jN`. A combination of the two, a cyclic distribution of chunks, is also available.

- dynamic scheduling - the iterations are again divided up into chunks, however as each processor finishes its allotted chunk, it dynamically obtains the next set of iterations.
- guided scheduling - is similar to static scheduling except that the chunk size decreases exponentially as each set of iterations is finished. The minimum number of iterations to be allotted to each chunk may be specified.
- runtime scheduling - this option allows the decision on which scheduling to use to be delayed until the program is run. The desired scheduling is then chosen by setting an environment variable in the operating system.

Extensive use of static and dynamic scheduling has been used in the HYDRA code.

4.4.2 Data Geometry

Particle-grid codes, of the kind used in cosmology, are difficult to parallelize efficiently. The fundamental limitation to the code is the degree to which the problem may be subdivided. Given that the computation-to-communication cost is relatively low compared with many other simulation and modeling codes (finite element codes for example), it must be ensured that the fundamental computational atom is sufficiently large that communication does not dominate the calculation and destroy the desirable scaling. For the HYDRA code, this corresponds to maintaining a low surface-area-to-volume ratio for the spatial blocks into which the particles and grid cells are decomposed. This sets an upper bound on the degree to which the problem can be subdivided, which in turn limits the number of processors that may be used effectively for a given problem size. The code is a good example of Gustafson's conjecture: a greater degree of parallelism may not allow arbitrarily increased execution speed for problems of fixed size, but should permit larger problems to be addressed in a similar time.

On SGI S^2MP machines, where data assignment across nodes may be specified, a block distribution for the global particle data arrays (*e.g.* positions, velocities) is chosen. Motivation for distributing the data is derived from the fact that a large amount of the computation proceeds via a linked list, which is ordered in the z-direction. For efficient memory access, the data organization must remain closely tied to the ordering of the linked list (or particle ordering).

As previously discussed, the code uses two meshes for force evaluation. The main Fourier mesh is used in the representation of the particle density data and a coarser chaining mesh is used to create a linked list of particles (or alternatively to provide a cell structure for the particle ordering) for the short range force. Typical sizes for the meshes are 256^3 and 116^3, respectively, for a simulation involving 2×128^3 particles. Memory limitations are not a significant issue with this code, as the storage requirements for the cubical arrays and the particle data amounts to 18N words for the serial code and 30N words for the parallel version due to local replication of data on PEs.

4.4.3 Particle velocity and position updates (updaterv)

This process can be parallelized trivially since the operations may be written as an addition of vectors (a direct consequence of this is that the operations 'vectorize', *i.e.* vector hardware computers can calculate this operation very efficiently). Parallelization is achieved simply by applying static scheduling with a chunk size set by the N divided by the number of PEs. A marginal complication arises because the cooling function uses a look-up table that must be updated. However it proved simple to assure that each thread does this just once.

4.4.4 Periodic mass assignment (mesh)

Mass assignment involves a race condition and thus does not parallelize trivially. The race condition occurs because it is possible for PE's to have particles which write to the same elements of the mass matrix. The approaches to solving this problem are numerous but consist mainly of two ideas,

1. selectively assign particles to PE's so that mass assignment occurs at grid cells that do not overlap, thus race condition is avoided

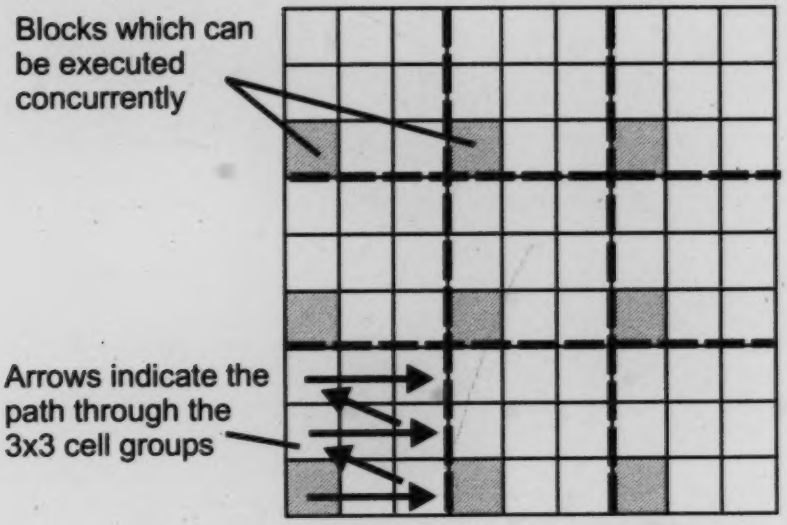

Figure 4.6: Cell grouping and sorting in the 1 × 1 load balancing scheme.

2. order the particles so that any contiguous block of particles has a definite upper and lower bound in one direction. Use slabs with ghost cells to accumulate the mass in each PE. Add up all the values in the ghost cells to accumulate the mass assignment array.

Ghost cells offer the advantage that they allow the calculation to be load-balanced but require more memory. Controlling which particles are assigned does not require more memory but may cause a load imbalance. Because the types of simulation performed have particle distributions that can vary greatly, both of these algorithms have been implemented.

Using controlled particle assignment

The particles in the simulation are ordered in the z-direction within the chaining cells. Hence, for a homogeneous particle distribution, if each PE is given a simple ordered block of particles to assign, of size N/number PEs, then the race condition is extremely unlikely to occur. This simple solution was adopted in early work (published in Thacker *et al.*, 1998), since it automatically provides good load balance. The solution is somewhat unsatisfactory since it does not remove the race condition. Tests in both clustered and unclustered states showed that, provided the number of PEs does not go above L/16, the race condition does not affect the results. However, this limitation clearly stops fine grain parallelism (explicitly limiting the number of PEs), and further highly inhomogeneous particle distributions, such as those encountered in multiple mass simulations, must use a different algorithm. The solution to this problem in the CRAFT implementation is to use the '*atomic update*' facility which is a lock-fetch-update-store-unlock hardware primitive that allows fast updating of arrays where race conditions are present.

A more elegant, and algorithmically correct approach (the race condition is absolutely removed) is to use the linked/ordered list to control the particle assignment. Since the linked list encodes the position of a particle to within a chaining cell, it is possible to selectively assign particles to the mass array that do not have overlapping writes. To assure a good load balance it is better to use blocks ($N \times A \times A$) of cells rather than slabs ($N \times N \times A$). Since there are more blocks than slabs a finer grained distribution of the computation can be achieved and thus a better load balance. For example, if all the particles resided in one slab, this would be a huge imbalance for a slab-based calculation, while the block-based calculation will still be able to distribute the blocks within the slab.

Chaining mesh cells have a minimum width of 2.2 potential mesh cells and Figure 4.1 displays a plot of the chain mesh overlaid on the potential mesh. When performing mass assignment for a particle, writes will occur over all 27 grid cells found by the TSC assignment scheme. Thus providing a buffer zone of one cell is not sufficient to avoid the race condition since a particle in cell one (assuming its x-coordinate is given by 2.199, just inside the right-hand boundary of the chaining cell at 2.2) may write to [2.199]+1=3, while a particle in cell two (with an x-coordinate 4.401, just

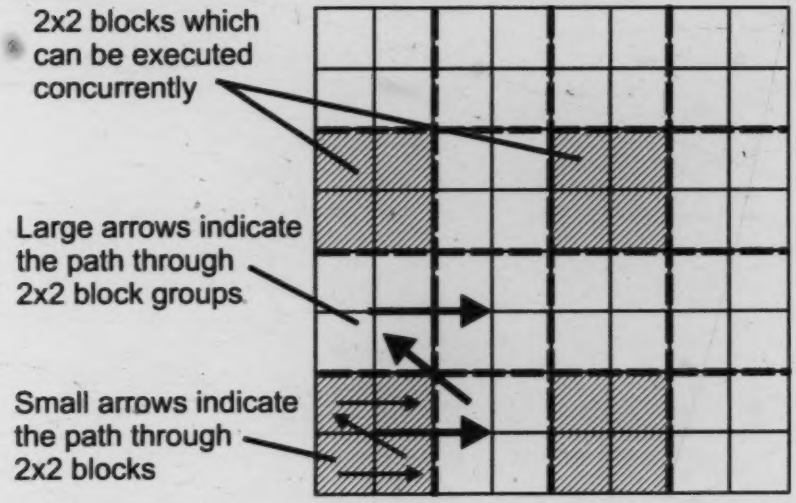

Figure 4.7: Cell grouping and sorting in the 2×2 configuration scheme.

inside the left-hand boundary at 4.4) may write to [4.401]-1=3. A space of at least 2.5 potential cells must exist between any two concurrent chaining cells involved in mass assignment. Thus between the nearest chaining cell that can be calculated and the current one, there must be a space of two cells. If the chaining cell were of width 2.5 cells, then one cell would be sufficient.

Because of this 'buffer zone' that must be accounted for, there are two distinct ways of performing the mass assignment using blocks:

- consider $N \times 1 \times 1$ blocks in 3×3 groups. Assign mass for particles in each of the block simultaneously and then perform a barrier synchronization at the end of each block. Since the blocks are in 3×3 groups there are nine barriers. See Figure 4.6 for a graphical representation of the algorithm.
- use $N \times 2 \times 2$ blocks which are grouped into 2×2 groups. Because each $N \times 2 \times 2$ block has a buffer zone of the same size as the $N \times 1 \times 1$ block the number of barriers is reduced to four (since the blocks are now in 2×2 groups). See Figure 4.7 for a graphical representation of the algorithm.

To avoid the possibility of load imbalance, a list of the relative work in each block (that can be evaluated before the barrier synchronization) is calculated by summing over the number of particles in the block. Note that the algorithm described here for preventing the race condition in mass assignment is also used in the short-range force calculation. In the short-range force calculation, calculating which blocks have the highest workload is a harder problem since there is no way of knowing the calculational load of each particle. The list does not make any allowance for particles in refinements since there remains the possibility that SPH particles may create a large amount of work. Future work will examine using a more sophisticated method of evaluating the workload of the blocks in the short-range force calculation. However, for the mass assignment scheme, simply summing the number of particles in each block provides an accurate estimation of the work in a block. Once the workload of each block has been evaluated, the list of relative workloads is then sorted in descending order. The calculation then proceeds by dynamically assigning the list of blocks to the PE's as they become free. The only load imbalance then possible is a wait for the last PE to finish which should be a block with a low workload. Static, and even cyclic, distributions offer the possibility of more severe load imbalance.

Tests show that on regular $2 \times N$ type simulations in all cases, the $N \times 2 \times 2$ blocks provide a better load balance than the $N \times 1 \times 1$ blocks, although there is only a very small (10%) difference between each method. Further, there is very little load imbalance because of the regular distribution of particles (as compared to that in a multiple mass type simulation). It is slightly surprising that the $N \times 2 \times 2$ blocks provide better load balancing since the $N \times 1 \times 1$ blocks offer finer granularity. However, they also require more barrier synchronizations and depend more heavily on the operating system to distribute the iterations (there are $(Ls/3)^2$ iterations per parallel section as compared to $(Ls/2)^2$ for the $N \times 2 \times 2$ blocks).

This method was not tested on multiple mass simulations, mainly because the particle density in a column which passes through all 4 particle-hierarchies will require $\propto N^3$ more operations than the column which passes only through the low particle-density region. There is thus a large potential load imbalance.

Particle ordering combined with slabs and ghost cells

This second approach is preferable since it provides a greater potential for achieving load balance, especially in multiple mass simulations. Particle ordering is achieved by sorting the particles in the z-axis within each linked list cell. This is *not* the same as simply ordering the particles by their z coordinate. The reason for sorting in this manner is that this particle organization is used in the ordered particle algorithm to speed up the calculation in the short range gravity calculation (see section 4.3.6). To achieve an efficient sort, *i.e.* one that is not affected by the underlying distribution being sorted, the HEAPSORT algorithm was used. HEAPSORT is preferable over other (potentially) faster algorithms like QUICKSORT since its run time is unaffected by the underlying distribution of particles. QUICKSORT can become slower than HEAPSORT for 'almost' ordered systems, which are likely to occur given that particles do not move far through chaining cells from step to step.

Achieving a complete parallelization of the sort is actually quite a difficult problem. Before the sort can be performed it is necessary to know how many particles are in each cell (the array `nhc`). From this list the entry points, denoting where in the particle list the cell begins, can be constructed (the array `ihc`). Calculation of the number of particles in each bin is performed in a loop over all particles in the serial code, and when parallelized it develops a race condition in the values of `nhc` since two PEs may have particles with the same cell index. This problem can be avoided by using ghost cells to pad the calculation of `nhc`. However, a more elegant alternative is possible. Consider the following: The cell index of all the particles can be calculated and stored in an array `ibox`. If `ibox` is then sorted in increasing order, using a parallel integer sorting algorithm, the returned array encodes the number of particles in each cell. This is because the list of particles now reads, for example, `1,1,1,1,2,2,2,2,3,3,4,4,4,4,4...` Thus to calculate `ihc`, this list can be sorted through in parallel and where the list changes from `i` to `i+1` is the `ihc` value for cell `i+1`. `nhc(i+1)` can then be calculated from `ihc(i+2)-ihc(i+1)`. Most importantly, all of these steps can be done in parallel. The key to making this algorithm efficient is having a fast parallel sorting routine. Fortunately a parallel sorting algorithm for shared memory machines is freely available and the performance is more than adequate (Li *et al.*, 1993; Mobarry and Crawford, 1996). Although originally written for Convex Exemplar machines it proved trivial to convert the code to OpenMP syntax.

The slab thickness used must account for the number of particles in the simulation, N, the size of the computational grid, L, the number of PEs, M, the width of the mass assignment function, $W_a = 2$, and the width of the chaining cells, $W_c \simeq 3$. The slab thickness, T, is calculated from,

$$T = \frac{N}{L^3}\frac{L}{M} + 2W_a + 2W_c + \text{padding} = \frac{N}{L^3}\frac{L}{M} + 10 + \text{padding} \tag{4.14}$$

The factor of $2W_a$ is included to allow for the width of the mass assignment scheme at the top and bottom of the cell (this is an overestimate since only half the mass assignment width sits of the edge of a slab). Similarly the $2W_c$ term allows for the limits of particle ordering within the chaining cells (the prefactor of 2 is necessary to account for the top and bottom of the slab). Thus the additional 10 cells + padding to account for the possibility of the particle distribution becoming very inhomogeneous, constitute the ghost cells. In multiple mass simulations, the higher particle density in the central region means that the slabs need to be thicker at the outer edges. This is compensated for by the N/L^3 term being greater than one. Note that this thickness is only an estimate of that required, hence the padding term. For example, pathological configurations, where all the particles reside at one point at the bottom of the grid, require a slab thickness of L. For the isolated solver it proved necessary to provide a very large amount of padding (more than 30 cells) since particles in refinements can become very centrally concentrated.

Once the slabs are set up, each PE assigns a block of particles of size N/M. This process has excellent load balance since each block is of the same size and must complete the same number of iterations. In practice tiny differences will occur due to the difference in cache performance for the particle distributions.

The summation of the slabs and their ghost cells into the potential mesh also involves a race condition. To avoid this problem, the maximum spatial extension in the z-axis of each slab is calculated while particles are being assigned. From this data an $M \times M$ 'concurrency matrix' is calculated that details which assignments can be performed in parallel because they do not write to the same potential mesh points. The matrix is calculated by checking for each slab i whether slabs $i+1$ to M overlap with i (including the effect of periodicity). If they do not overlap, 1 is entered in

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0\downarrow & 0 & 0 & 0 & 0 & \dots & 0 \\ 1^{\dagger}\rightarrow & 0\rightarrow & 0\downarrow & 0 & 0 & \dots & 0 \\ 1 & 1 & 0\downarrow & 0 & 0 & \dots & 0 \\ 1 & 1 & 1^{\dagger}\rightarrow & 0\rightarrow & 0\downarrow & \dots & 0 \\ 1 & 1 & 1 & 1 & 0\downarrow & \dots & 0 \\ 1 & 1 & 1 & 1 & 1^{\dagger}\rightarrow & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 1 & 1 & 1 & \dots & 0 \end{bmatrix}$$

Figure 4.8: Example of a search through the concurrency matrix to determine which loads can be performed concurrently. Those slabs that can are identified with a dagger.

the array, otherwise the entry is zero. The slabs that can be assigned concurrently are then found by sorting through the concurrency matrix by starting at the column of the first slab and searching vertically until the first 1 is found. Once found the search moves across to the column corresponding to the row in which the 1 was just found and then the search proceeds downward again until a 1 is reached. This process continues until all slabs have been identified. An example of this search is shown in Figure 4.8.

Because all the slabs cannot be loaded at the same time, a loss of efficiency occurs. However, since there is comparatively little work in the assignment of the slabs to the potential mesh as compared to the calculation of the mesh values in the slabs, this does not affect performance significantly. In Figure 4.9 an expected value of the parallel scaling of the algorithm, for a fixed 128^3 mesh and particle number, is plotted. This plot incorporates the serial overhead involved in searching through the concurrency matrix and assumes *perfect parallel efficiency in the remaining sections of the algorithm*. Clearly for large number of PEs there is a significant fall off as the amount of work per PE becomes smaller and the relative amount of effort in the serial search becomes higher.

4.4.5 Fast Fourier transform (four3m)

Since the 3-d FFT is broken down into a series of 1-d FFT's, it is comparatively simple to parallelize. For example the first set of FFT's (in the z-direction) are distributed across the machine, next the y FFT's and finally the FFT's in the x-direction are performed. The same analysis applies to the inverse transform.

4.4.6 Periodic Fourier convolution (cnvl)

Although the planes involved in the convolution are independent, the serial code was written in such a way that the outermost loop could not be parallelized over. However, the work within a plane can be parallelized. Hence rather than rewriting the code it was decided to use the inner parallelization. This method is still quite effective since there are roughly $(L/2)^2 * 100$ operations to be distributed, *i.e.* for a 128^3 mesh 409600 operations have to be divided among the PEs. The factor of 100 is an estimate of the number of calculations at each grid point. Even for 128 PEs, that still leaves 3,200 operations per PE which is still higher than 1,000 operations per PE limit which SGI quotes as the minimum number of operations per thread to achieve parallel speed-up (SGI, 1997).

4.4.7 Force interpolation (mesh)

The force array is calculated from the potential array using a 10-point finite differencing operator (Couchman, 1992). Since this only involves a read operation from the potential array, and a write to independent points of the force array, this operation vectorizes. The loop over the force array entries is performed using three nested loops, and hence, parallelization is performed across the outermost loop. These three loops can be written as one, allowing finer subdivision of the operation if necessary. The accumulation of the particle accelerations also vectorizes since it involves reads from the force array and independent writes to the acceleration array. Thus parallelization is achieved by distributing the loop over all the particles.

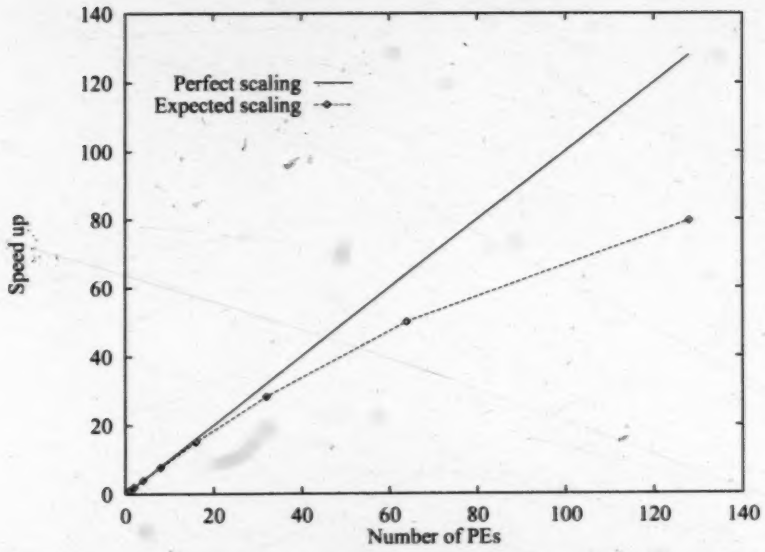

Figure 4.9: Maximum parallel scaling possible for the parallel mass assignment algorithm which uses slabs and ghost cells. The scaling is limited by the serial operations used in searching through the concurrency matrix, which then determine the scheduling of the parallel threads. In principle, the mass assignment scheme which uses ordered particles and barrier synchronization can scale almost perfectly, although in practice it will be limited by imbalances due to the particle distribution.

4.4.8 Pair-wise forces (shgravsph)

The short range forces are accumulated by using 3 nested loops to sort through the chaining mesh. The outermost loop controls the z-iterations and if this loop is distributed across the PEs then a slab decomposition results. Note that a race condition arises when this simple approach is taken (neighbouring PEs can write to the same particle data) but since the work quantum is large and evenly distributed *provided that the particle distribution is comparatively homogeneous*, it should not be a significant problem. Furthermore, instead of using particle indices to schedule the computation, it proceeds over the chaining cells which have a fixed geometry. The first solution for parallelizing this routine was to simply schedule the outermost loop across the PEs (Thacker *et al.*, 1998). In the tests conducted, no failure was found provided that the number of PEs was less than Ls/8, where Ls is the size of the chaining mesh along one dimension. Note that if a barrier is placed at the half way point of each slab, then the race condition is completely removed, although this was not implemented. The race condition is avoided in the CRAFT version by the use of the atomic update primitive. However, the problem with this algorithm is the same as in the first method used for mass assignment: the number of PEs is limited and the algorithm will fail with inhomogeneous particle distributions.

The main problem with a slab decomposition, which was discussed in section 4.4.4, is that it does not exhibit sufficient granularity to provide a good load balance in clustered systems. The block decomposition, on the other hand, does provide sufficient granularity. The race condition can be avoided in a similar fashion to mass assignment. In this case, it is only necessary to consider the fact that any one cell will write to the particle data in all 26 neighbour cells. Hence a buffer of two cells must be provided between cells which can be calculated at the same time. Tests showed that of the two possible block sorting algorithms discussed in section 4.4.4, $N \times 2 \times 2$ blocks are more efficient than the $N \times 1 \times 1$ blocks. The difference in execution time in unclustered states was negligible, but for highly clustered distributions (as measured in the Santa Barbara cluster simulation), the $N \times 2 \times 2$ method was approximately 20% faster. This difference is partially attributable to the difference in the number of barrier synchronizations required by each algorithm: the $N \times 2 \times 2$ require four while the $N \times 1 \times 1$ require nine. Also the $N \times 2 \times 2$ blocks have better cache reuse than the $N \times 1 \times 1$ blocks since particle data used by the first cell will also be used by the second, third and fourth cell in the same z plane of the block. Reuse also occurs due to cells below the initial z plane requiring the same particle data.

4.4.9 Refinement placing (refine)

As already discussed in the review of the serial implementation, placing of refinements is a very complex optimization problem. Research into parallelization of this routine is currently ongoing with Dr. Stephen Booth at the Edinburgh Parallel Computing Center.

As an interim measure, it is simple to force the placement refinements over large meshes to occur only every 10 steps. Since the large scale distribution does not change significantly over 10 steps (the time-step limitation from the velocity assures that $dt \leq 0.4\epsilon/|\mathbf{v}|_{max}$), almost no inefficiency in refinement placing occurs as a result. This approach was adopted in the CRAFT implementation. The alteration is quite simple since all that is required is a check of the iteration number and the size of the grid which is being examined. The hierarchical nature of refinement placing ensures that this procedure cannot go wrong (for example a 64^3 refinement will not be placed inside a 32^3 refinement).

Once refinements have been placed it is necessary to calculate how many particles are contained within the refinement. This involves a loop over the chaining cells in a refinement. Provided that the cells do not overlap, this step can be done in parallel. For simulations with manual refinement placing, this is not always guaranteed, since manual refinements are specified after the refinement placing algorithm has been called, thus in such cases this step is left unparallelized.

4.4.10 Distribution of refinements (force)

The adaptive part of the algorithm is managed in two ways. Firstly, the algorithm may place sub-meshes that have a Fourier grid dimension of 64^3 or greater. If so, the calculations for these refinements are performed in parallel across all of the PEs. This avoids a possibly serious load imbalance that could occur if one very large refinement was placed and calculated on a single PE. However, the 64^3 refinements are inefficient when done across all the PE's. A partial cause of this is the reuse of larger arrays which have been declared using a block decomposition. For example, if an array of size 16,000 is block-decomposed across 8 PE's, then each PE will be responsible for storing 2,000 elements. In the event that only 2,000 elements are used, and the calculation is spread across all available PEs, a major bottleneck will occur at the memory of the first PE. Note that in actual fact, 2 PEs share one memory board, so the decomposition is strictly given by $N/(2 \times PE)$. The access problem can be partly alleviated by using a block cyclic distribution of data, however in this case the efficiency becomes poor if the block size is significantly smaller than the minimum page size, 16 kB. As a compromise, for the large isolated refinements, a block cyclic distribution of the arrays is used, with a block size of 1024 elements (one quarter that of the memory page size).

The smaller sub-meshes ($L \leq 32$) are distributed as a task farm amongst the PEs. As soon as one processor becomes free it is immediately given work from a pool. Guidance for this process is provided by the load scheduler in the compiler (the OpenMP dynamic scheduling option). By using dynamic scheduling the possibility of a load imbalance is decreased.

Load imbalance occurs in the task farm if one refinement takes longer than the rest and there are not enough refinements to balance the workload. The imbalance is more likely to occur when the ratio refinements to PEs is low. This is a significantly problem for the CRAFT code where the number of PEs may be as high as 512. This problem emphasizes one of the drawbacks of a shared memory code—it is limited by the parallelism available and has to choose between distributing the workload over the whole machine or single CPUs. It is not possible in the SMP programming environment to partition the machine into tailored subunits. This is the major drawback which is being addressed by the development of an MPI version of the code.

4.4.11 Isolated mass assignment (imesh)

It is important to note that the isolated solver will usually be applied to particle distributions that are strongly centrally concentrated (since refinements are placed over such regions). Thus, it would be expected that the algorithm that uses controlled particle assignment (as described in section 4.4.4) could suffer a serious load imbalance. Hence the particle ordering plus ghost cell algorithm (section 4.4.4) is used.

4.4.12 Isolated Fourier convolution (icnvlt)

The isolated Fourier convolution is slightly more difficult to parallelize than the periodic version since the serial code was optimized as much as possible. The first section of work involves Fourier transforming lines parallel to z axis. Since the lines are sorted into a temporary buffer, each PE is given its own buffer. The load is well balanced, since the same number of calculations are applied to each PE and hence simple chunking of the iterations is used. Once the lines have been FFT'd, the results are deposited into a temporary 3-d array during the parallel loop. Once this step is complete

the rest of the Fourier transform (*i.e.* in the x & y directions) is performed as part of the same loop that controls the green function convolution in Fourier space and also the inverse transform in the x & y directions. Two planes are calculated per iteration of the loop which means that for isolated 64^3 FFTs, the fastest calculation is attained with 32 PEs. Adding more does not produce any speed-up. This routine could be rewritten to avoid this problem but since the convolution is not particularly time-consuming, it was decided not to do this. As in the first loop, since the same number of calculations are used per plane, simple chunking of the iterations is adequate to ensure good load balance. Once the loop over planes is complete, the inverse transform in the z-direction must be performed. This step uses the same parallel scheme as the forward transformation.

4.4.13 Isolated force interpolation (imesh)

same The calculation of the force array for the potential field uses the 10-point differencing operator used in the periodic solver. Hence the same parallel scheme can be applied without any modification. This also applies to the calculation of the acceleration values.

4.5 Performance of the parallel code

A number of numerical experiments were conducted to test the scaling of the parallel code. Since the scaling characteristics of the code can be notably different under clustering the performance data from the code was examined at both high and low redshift.

For benchmarking purposes it was decided to create the simulation cube by tiling the 2×64^3 'Santa Barbara' cluster simulation to generate a larger data set. This corresponds to increasing the box size, rather than increasing the linear resolution of the simulation. Figure 4.10 compares the increase in the box size and the relative clustering within the different simulations. One minor drawback of this method is that it forces the largest refinement that is placed to be smaller than the size of the replicated tile.

4.5.1 Simulations with homogeneous particle distributions

Table 4.1 displays the parallel speed-ups and efficiencies for a number of different combinations of PEs and problem sizes. Results up to 16 PEs were run in multi-user mode and may include inefficiencies due to overheads from the kernel scheduling other jobs. Also note that very large variances in the execution time were seen, depending upon whether the environment variable `MPC_GANG` was set to `ON` or `OFF`. When turned on, the kernel is forced to give each PE a similar amount of execution time on each processor. When turned off, the PE must compete with other jobs for time on any processor. Hence, as would be expected, the scaling was observed to be much better when `MPC_GANG` was set to `ON`. The largest observed difference was a reduction in run time from 40 seconds to under 30 (*i.e.* a 25% performance increase), in the 8 node 2004 run.

The results for the 128^3 mesh in the unclustered state are very encouraging. The code maintains a reasonable parallel efficiency (63%) up to 32 PEs. Given these results, it would be expected that the 256^3 mesh should scale even better, since a larger work quantum is available to each PE. Unfortunately, this was not the case. The reason for this is not immediately obvious, but may well be due to memory placement effects. As the code is written, a lot of memory reuse is obtained by overwriting a large common block denoted `WRK`. No data geometry is initially assigned to this common block, and the kernel is allowed to initialize the memory at the memory node next to the first CPU to reference the memory address ('first touch' memory placement). Since the parts of the code that reference the common block require different memory placement to work efficiently, a significant slow down occurs. The clearest example of this was the **list_p** routine which utilized the `WRK` array as calculation space for the **psrssort** parallel sort algorithm. In this form sorting 2×256^3 particles took 23 seconds for 32 PEs, while declaring the calculation space separately from the `WRK` common block (at a cost of $2N$ memory words) reduced the time to 2.55 seconds - *almost a factor of 10 improvement.* This problem is a cause for concern since, if this kind of slow-down can occur due to poor memory placement, the overall scaling can be reduced drastically. Comparing the 256^3 results to those from 512^3 does show improvement for the larger problem size (speed-up for 128 PEs is 41.5 for 512^3, versus 32.9 for 256^3).

To further understand the performance characteristics of the code, execution times for important parts of the calculation were analysed separately. Figure 4.11 shows a breakdown of the execution time of the code for the 128^3 mesh in the unclustered state, in terms of the time for each subroutine. For 1 PE the execution time is dominated by **mesh** and **shgravsph** (which calculates the short range force). Increasing to 2 PEs shows a consequent reduction for all the calculation parts. Note

Run	N	Mesh	PEs	Redshift	Wall clock	Speed-up	Efficiency
1001	2×64^3	128^3	1	19	19.3	1.00	100%
1002	2×64^3	128^3	2	19	10.3	1.87	94%
1003	2×64^3	128^3	4	19	5.85	3.30	83%
1004	2×64^3	128^3	8	19	3.14	6.15	77%
1005	2×64^3	128^3	16	19	1.66	11.6	73%
1006	2×64^3	128^3	32	19	0.96	20.1	63%
1011	2×64^3	128^3	1	0.001	83.4	1.00	100%
1012	2×64^3	128^3	2	0.001	44.4	1.88	94%
1013	2×64^3	128^3	4	0.001	25.9	3.22	81%
1014	2×64^3	128^3	8	0.001	17.6	4.74	59%
1015	2×64^3	128^3	16	0.001	14.5	5.75	36%
1016	2×64^3	128^3	32	0.001	12.6	6.62	21%
2001	2×128^3	256^3	1	19	170	1.00	100%
2002	2×128^3	256^3	2	19	88.3	1.93	96%
2003	2×128^3	256^3	4	19	51.3	3.31	83%
2004	2×128^3	256^3	8	19	28.3	5.86	73%
2005	2×128^3	256^3	16	19	20.5	8.29	52%
2006	2×128^3	256^3	32	19	11.6	14.7	46%
2007	2×128^3	256^3	64	19	7.62	21.8	34%
2008	2×128^3	256^3	128	19	5.05	32.9	26%
2011	2×128^3	256^3	1	0.001	610	1.00	100%
2015	2×128^3	256^3	16	0.001	78.7	7.75	48%
2016	2×128^3	256^3	32	0.001	56.0	10.9	34%
2017	2×128^3	256^3	64	0.001	52.7	11.6	18%
3006	2×256^3	512^3	32	19	79.6	(17.9)	(56%)
3007	2×256^3	512^3	64	19	57.6	(24.8)	(39%)
3008	2×256^3	512^3	128	19	34.4	(41.5)	(32%)
3017	2×256^3	512^3	64	0.001	347	(12.9)	(20%)

Table 4.1: Overall parallel scaling for simulations with comparatively homogeneous particle distributions. The simulations were run on an 128 node (300 Mhz R12000) Origin 2000 system at the SGI-Cray Eagan Supercomputing Centre. Values in parenthesis are estimated.

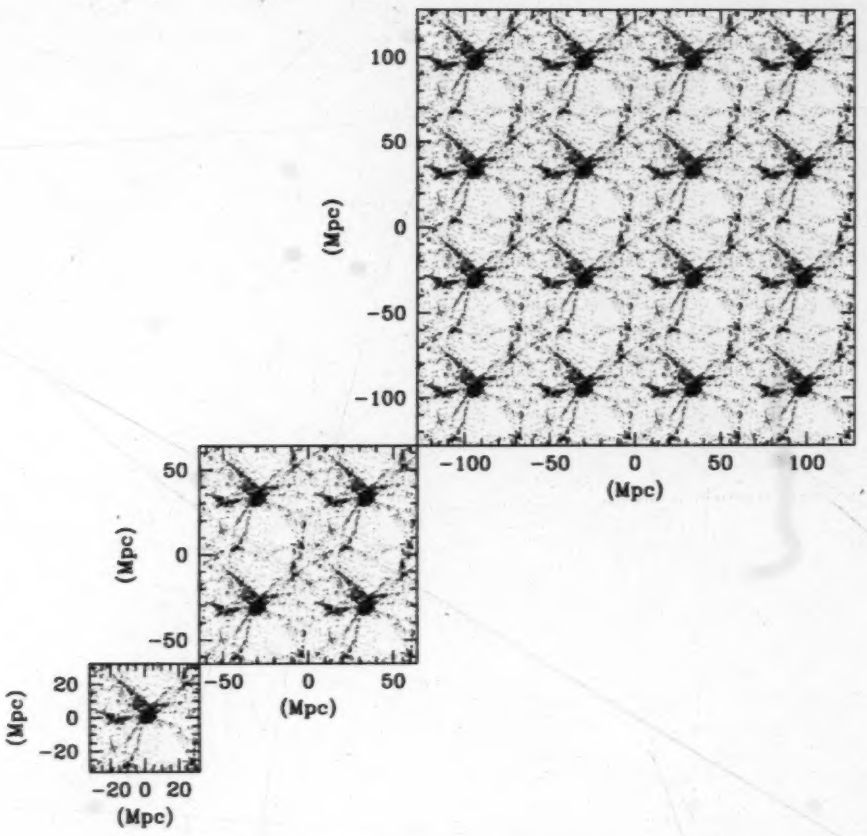

Figure 4.10: Relative scaling of the test simulations. The cluster simulation is replicated either 8 or 64 times to form the larger simulations. The diagram shows a progression from 64^3 to 256^3, albeit projected on to 2-dimensions.

increasing from 2 to 4 PEs shows an *increase* in the list creation time. This result is due to the poor memory placement problem mentioned previously, although for smaller problems the effect is not as severe. Overall, the breakdown shows that in the unclustered state no part of the algorithm breaks the parallel scaling.

In Figure 4.12, the execution time in the clustered state (128^3 mesh) is decomposed into the time to calculate the top level grid, the large level 1 refinements, calculated across the whole machine, and the time for the refinement farm. Although the terminology 'large level 1 refinement' is used, strictly speaking these refinements may occur on any refinement level. For a small number of PEs (less than 8), there is little difference in the relative breakdown of the execution time. For 8 PEs and beyond, the time for the large level 1 refinements does not reduce, while the top level grid and refinement farm continue to show at least some reduction in time. Parallel scaling in the clustered state is less efficient for a number of reasons. Firstly, there is a potential for load imbalance in the parallel calculation of the short range gravity force since one block may take significantly longer than another. The clustering of particles also causes potential problems in list creation and particle reordering since cells may contain significantly different numbers of particles. However, probably the main cause of the reduction in scaling, as has already been hinted at, is the performance of the large level 1 refinements. These refinements have a mesh size of 64^3 and above, although in the simulations conducted only 64^3 meshes were generated—which is the most inefficient configuration

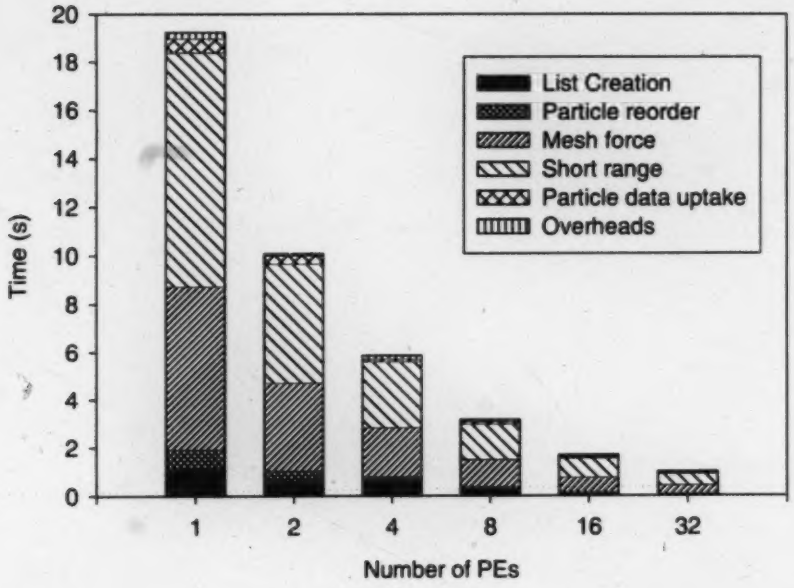

Figure 4.11: Execution time in seconds for the most computationally expensive subroutines versus the number of PEs for the unclustered 128^3 simulation. If any one routine was not showing significant speed-up, and hence impairing the overall code speed-up, it would be evident in this plot.

possible. The maximum speed-up observed for these refinements is only a factor of 3, hence the maximum speed-up of the total calculation time of large level 1 refinements is 3. The degradation in the scaling is clearly seen in table 4.1, where for the 128^3 mesh and 32 PEs the speed-up falls from 20.1 to 6.62.

Since ultimately the code will be used for conducting larger simulations, rather than performing standard size simulations faster, it is of value to study the performance for larger problems in detail. Hence in figures 4.13- 4.16, the speed-up for the top level routines **shgravsph**, **mesh**, **list** and **reorder** are shown for the 256^3 and 512^3 meshes and a variety of PEs. Note that since the 512^3 problem is too large to run even a single time step on 1 PE, the speed-ups were estimated by scaling the 256^3 results.

The parallel scaling of **shgravsph** is interesting since it has a very large workload in the clustered state (in the serial code it may be as much as 60% of the execution time). Results for the large simulations are shown in Figure 4.13. The unclustered 256^3 simulation shows a sub-linear speed-up, but does at least continue to improve performance up to 128 PEs. Surprisingly the clustered 256^3 simulation has better scaling than the unclustered state up to 32 PEs, after which it falls to the same as the unclustered. The unclustered 512^3 and 256^3 simulations show virtually identical speed ups for 64 and 128 PEs.

In Figure 4.14 the scaling for **mesh** is shown (note that this is the ordered particle assignment scheme of section 4.4.4). In the unclustered state mesh accounts for approximately 20% of the run time, although this falls as the system becomes clustered since more work added to the short range force. Remarkably the observed speed-up seems to be comparatively constant, whether the simulation is clustered or not. Note that this may be partially a consequence of the clustered state being created from tiling a smaller simulation. The fact that the algorithm does not scale particularly efficiently (approximately 30% for 128 PEs) is probably caused by the dynamic allocation of the threads - since the memory must be moved to whichever PE requires it. The overall scaling is somewhat lower than that predicted for the ghost cell algorithm.

Figure 4.15 displays the speed-up of the list creation algorithm. Of all the algorithms analysed (other than **updaterv** which scales trivially since it is a vector operation) this one shows the best results. The scaling for the unclustered 512^3 simulation is truly excellent, with speed-up on 128

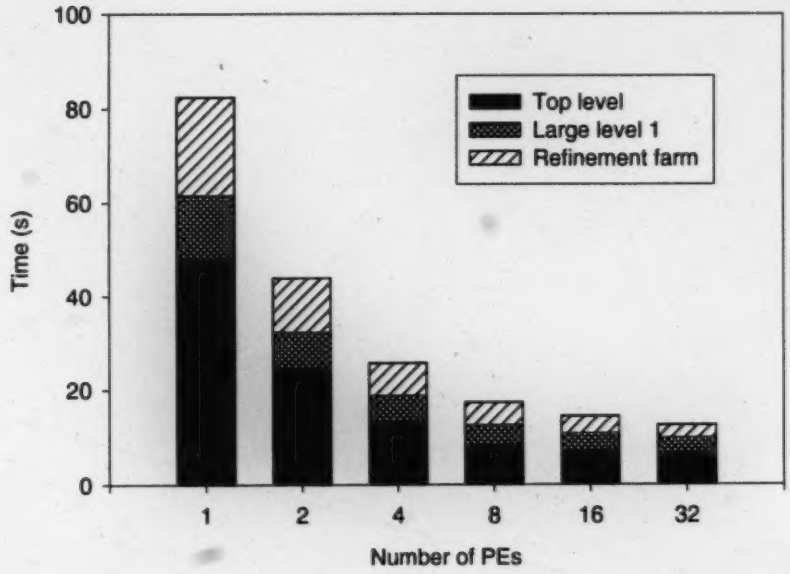

Figure 4.12: Scaling of the top level grid, large level 1 refinements and refinement farm at the end of the 128^3 simulation. The large level 1 refinements scale up to about 4 PEs and then performance increase stops abruptly, leading to a limit on the maximum scaling.

PEs being 94. For the unclustered 256^3 simulation the scaling is good up to 64 PEs (50), but falls off at 128 PEs. The clustered state is less efficient which is caused by a load imbalance due to the chaining cells having very different numbers of particles. Notably, the clustered 512^3 simulation shows a drastic reduction in efficiency—the cause of this is unclear. However, it is not a major cause for concern since list creation still only constitutes 6 seconds in the total 347 second execution time.

The final subroutine examined in detail is **reord** which reorders the particle data using the list created in **list**. Results are plotted in Figure 4.16. The algorithm is quite simple: particle data is first loaded into a temporary array, using the list created in **list** to order it in the correct way. The temporary array is then used to write the reordered particle data back into the original array. The scaling for the unclustered 256^3 and 512^3 simulations show sub-linear scaling beyond 32 processors, with the 256^3 simulation achieving a maximum speed-up of 40, and the 512^3 simulation 57. The clustered state is less efficient, and the 512^3 simulation shows almost the same scaling as the unclustered 256^3 simulation. The clustered 256^3 shows only slightly less scaling than the unclustered state, and for 32 PEs speed-up is slightly below 20.

In the final plot for homogenous particle distributions, Figure 4.17 shows the relative time of the top level grid, large level 1 refinements and refinement farm. This diagram shows very effectively how the lack of scaling in the large level 1 refinements limits the parallel speed-up. For the 512^3, simulation the large level 1 refinements constitute slightly over 60% of the run time, while for the single PE 256^3 simulation they constitute only 10% of the time. The same applies to the 32 and 64 PE 256^3 runs, where the large level 1 refinements account for roughly 50% of the execution time. Ways of circumventing this problem are discussed in the conclusion.

4.5.2 Simulations with inhomogeneous particle distributions

Multiple mass simulations are particularly difficult for mesh codes to cope with. The fundamental problem is that the particle distribution begins in a clustered state, and then evolves into an even more clustered one. For AP^3M, effective treatment of this particle distribution requires hand placement of the large level 1 refinements since the refinement placing algorithm is designed to cope with

Run	N(base)	PEs	Redshift	Wall clock	Speed-up	Efficiency
1001	32^3	1	67	16.4	1.	100%
1002	32^3	2	67	9.92	1.65	83%
1003	32^3	4	67	7.80	2.10	53%
1004	32^3	8	67	6.71	2.44	31%
1011	32^3	1	0.78	70.8	1.	100%
1012	32^3	2	0.78	66.7	1.06	53%
1013	32^3	4	0.78	62.3	1.13	28%
1014	32^3	8	0.78	57.0	1.24	16%
3002	100^3	2	67	258	(1.88)	(94%)
3003	100^3	4	67	138	(3.51)	(87%)
3004	100^3	8	67	78.9	(6.15)	(77%)
3005	100^3	16	67	54.4	(8.92)	(56%)
3006	100^3	32	67	32.0	(15.2)	(48%)
3012	100^3	2	2.48	1090	(1.88)	(94%)
3013	100^3	4	2.48	807	(2.54)	(64%)
3014	100^3	8	2.48	712	(2.87)	(36%)
3015	100^3	16	2.48	690	(2.97)	(19%)
3016	100^3	32	2.48	650	(3.15)	(10%)
4016[a]	100^3	32	2.16	1980	-	-
4017[b]	100^3	32	2.16	135	-	-

Table 4.2: Overall parallel scaling for simulations with inhomogeneous particle distributions. Note that these simulations include radiative cooling and can form very dense gas regions. Values in parenthesis are estimated. The simulations were conducted on the UK-CCC 'COSMOS' server, which is an Origin 2000 system with 44 250 Mhz R10000 CPUs. [a]Run 4016 contains the contribution from the star formation algorithm, of order 200 seconds. [b]Run 4017 is a dark matter only run designed to show what scaling is possible when the inefficiency in the SPH algorithm is removed.

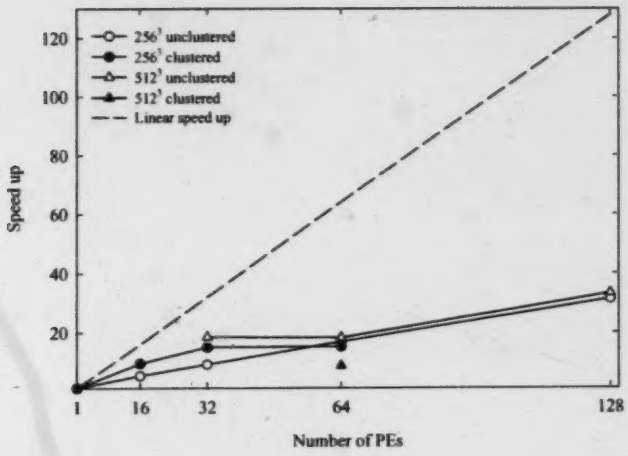

Figure 4.13: Parallel speed-up for the top level **shgravsph** routine. Although the speed-up is far from ideal performance does at least continue to improve up to large number of PEs.

clustering in a single level simulation.

Table 4.2 shows the results from both a 32^3 layered simulation and a larger 100^3 simulation. The smaller simulation does not scale particularly well and since the potential is found on a 64^3 mesh this is to be expected. Previous sections have shown that for this size mesh, the maximum speed-up is about a factor of 3. By $z = 0.78$ the scaling falls off markedly, due to two factors: (1) one of the columns in the high resolution refinement takes far longer than the others since it contains particles with almost $100N_{smooth}$ neighbours which causes a P^3M-like slow-down and (2) there are only 4 refinements in total, two of which are done across the entire machine.

For the 100^3 simulation the findings are similar, although the parallel efficiency for the initial conditions is really quite reasonable, 48% for 32 PEs. As soon as clustering develops scaling is lost for the same reasons as discussed in the previous paragraph (although for this case the highest number of neighbours is about 15,000, which increases to over 30,000 by $z = 2.16$). This time the large refinement takes almost 430 seconds to calculate, almost of all of which is taken up by the cost of the single column laying over the very high resolution region. It is unclear why this region was not calculated in a further refinement since most of the particles lying within it should have h values close to h_{min}. There are more refinements in this simulation though, over 150 are placed, and 90% are small enough to place in the refinement farm. To estimate the effect of the slow-down due to the imbalance in the SPH algorithm, a dark matter only simulation was carried out by converting all gas to stars. The resulting dark-matter only run is *over ten times faster* than the SPH+dark matter calculation.

4.6 Summary and discussion

It has been demonstrated that shared memory systems, accompanied by efficient shared-memory compilers, provide a simple programming model that allows rapid development of parallel codes. This is due to the global array addressing provided by the shared memory model negating the need for explicit message passing. Consequently, the programmer can focus more on the development of the algorithm rather than doing so in concert with a data distribution algorithm. Even so, parallelization of gravitational particle-in-cell codes is difficult because of the extreme distortion of the map between the Lagrangian particle representation and the Eulerian mesh.

The SGI Origin 2000 series of supercomputers are a powerful platform to conduct simulations

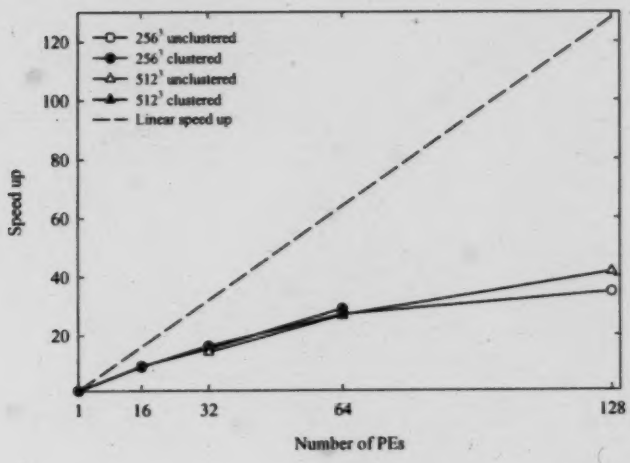

Figure 4.14: Parallel speed-up for the top level **mesh** routine. The scaling is similar regardless of particle number or mesh size, suggesting that memory bandwidth may be a limiting factor.

on. The increased memory bandwidth inherent in the NUMA architecture overcomes the traditional bottlenecks associated with shared memory systems. Careful tuning of data placement is necessary to avoid unnecessary slow-down in execution speed and to utilize the large bandwidth available on the machine. However, it should be noted that, when compared to the effort of parallelizing the code as a whole, only a small amount of effort is required to decide upon data placement. The OpenMP API, which now runs on almost all SMP platforms, provides a very flexible shared memory programming model. By utilizing a number of different iteration scheduling options load balance problems can be reduced significantly.

Even before parallelizing a code, optimization of the algorithm for the architecture of the CPU can have significant performance improvements. It was demonstrated that by substituting a linked list with an ordered list of particle indexes, better cache utilization on RISC CPUs can be achieved. Further, ordering particles within cells, so that the list structure is removed entirely, proved to be even more efficient. For this method, cache-reuse is optimized and the cache-miss ratio is very small. A performance gain of almost two and a half times the linked list routine is possible.

The most conceptually difficult part of the parallelization is caused by the refinements. A distinction has to be made between refinements which can be distributed across the whole machine or alternatively those which are performed as a task farm. Since the decision of whether to treat the refinement as parallel or not is dependent upon the size of the Fourier grid, it was possible to use conditional parallelism on the isolated solver routines. This negated the need to write both parallel and serial isolated routines.

Coping with the race conditions in mass assignment and the short range force calculation is nontrivial. While 1-d decompositions of the calculation are simple, they quickly lead to inefficiencies as clustering of the particles develop. This necessitates the use of 2-d distributions, which, by utilizing dynamic distribution of the calculation, have been shown to be extremely effective at providing a good load balance. Coping with the race conditions also leads to complications in the distribution of blocks used in 2-d distribution. To avoid particle data being written to at the same time, buffer zones and barrier synchronizations are used.

A large part of the parallelization is straightforward. In particular, the particle updates of position and velocity (plus hydrodynamic-related quantities) can be done by using simple division of the iterations across PEs. Also the parallelization of the Fourier transform and convolution routines was comparatively straightforward. Since the number of calculations is constant, load balancing is not a problem and the same distribution of iterations can be used at each time step. The highest

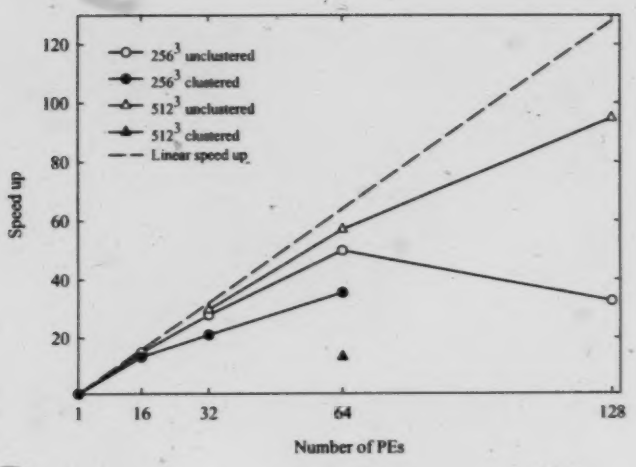

Figure 4.15: Parallel speed-up for the top level **list** routine. A degradation occurs in the clustered state since cells can have widely different numbers of particles, further it is unclear why the clustered 512^3 shows such a large performance decrease relative to the unclustered state (given that the 256^3 simulation does not show such a decrease). However, both of the unclustered simulations show excellent scaling up to 64 PEs.

efficiency is achieved in these parts of the code since the iteration distribution is closely tied the data distribuion.

Almost all of the routines with a large work quantum were parallelized. The only exception was the refinement placing algorithm, which is a particularly difficult problem to parallelize. A simple work-around of performing refinement-placing every ten steps was used. For the comparatively homogeneous particle distributions found in fixed resolution simulations, the code speed-up was good. Degradation of the scaling occurred due to placement of refinements which are too small to be calculated across the whole machine efficiently, but are too large to calculate on an individual PE. A possible solution to this problem is to increase the size of the smallest refinement calculated across the machine, although this would increase memory requirements signficantly since each PE in the refinement farm would require storage for up to 262,144 particles. Scaling for a 2×256^3 simulation at the $z = 0$ epoch (worst case) was 20% on 64 nodes of an Origin 2000 system, and the cycle time was 357 seconds. It is clearly possible to perform these simulations within a reasonable wall-clock time frame (*i.e.* approximately 4 days on a dedicated server).

For simulations with inhomogeneous particle distributions, such as the multiple mass technique, scaling was good initially but fell off sharply under clustering. The primary cause of the problem is very dense regions of SPH particles which have a very high workload. The parallel granularity is not sufficiently small to cope with this problem, especially considering that the size of these regions is far smaller than that of the chaining mesh. Three possible approaches to resolving the problem are to,

1. Change the algorithm to find cells with potential problems like this and perform the entire cell calculation across the machine (requires significant rewriting).

2. Remove the SPH calculation from the **shgravsph** routine and incorporate it into an alternative solver where the list structure can be more finely decomposed (*e.g.* use a tree).

3. Change the short range behaviour of the SPH algorithm so that the particles always smooth over approximately N_{smooth} particles.

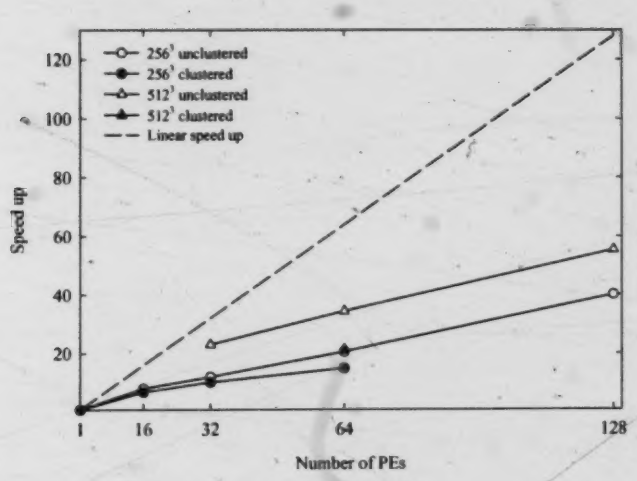

Figure 4.16: Parallel speed-up for the top level **reorder** routine. Speed-up is sub-linear, but again, does occur to a large number of PEs.

The third option is the simplest to try. The dark-matter only simulation, which continued to show speed-up in the clustered state, gives a hint of the possible performance. Note that given the code still provides some speed-up, using it is worthwhile. The scaling problem does not occur to such a significant degree in the CRAFT code since the short range calculation is performed over a cell-by-cell 3-dimensional decomposition of the chaining cells. Note that the cells are ordered in the same fashion as the 2-dimensional blocks used here, *i.e.* those of highest work are considered first. However since there is no barrier synchronization (because of the hardware locking available on the T3D), a single thread can spend all its time on the cell without hindering the progress of others—in contrast to the work presented here where barrier synchronizations must be performed.

It remains the case that grid-based particle codes are the most efficient for conducting simulations of periodic regions of the Universe. Tree codes cannot compete in this regime, since not only must images be calculated to provided a periodic force, but also the $N \log N$ coefficient of the method is that much larger than the one associated with grid based methods. As a result, grid-based codes are roughly one order of magnitude faster than tree codes on these problems. It should be emphasized again that although P^3M suffers a severe slow-down under clustering, AP^3M can be used to avoid this problem. This may not be true for inhomogeneous distributions such as that encountered in multiple mass simulations (at least with the HYDRA code in its current form). The main cause of this problem is that the algorithm currently performs the SPH and gravity calculation in the same subroutine, this can lead to significant slow-down, since the SPH calculation for a particle may be far more time consuming than the gravity. Since tree codes are relatively unaffected by the particle distribution (compared to the slow-down observed in table 4.2) they may well be better at calculating the evolution of these systems. However, a number of suggestions for improving the HYDRA algorithm have been made, that should help to alleviate the load imbalance. Further, the dark matter only calculation still showed good scaling in the clustered state. Hence, it remains to be seen whether tree codes really do have a significant advantage in these simulations.

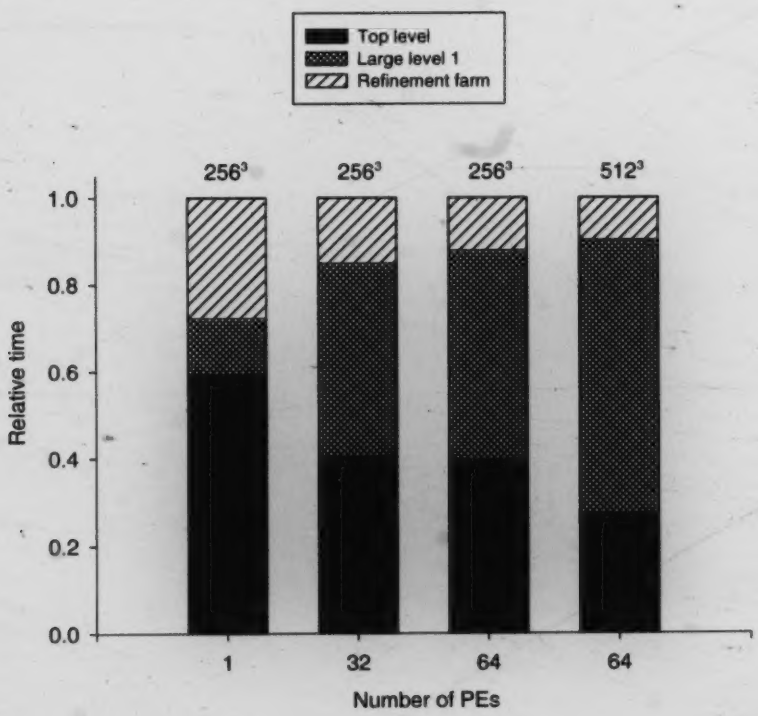

Figure 4.17: Relative cost of the top level grid, large level 1 refinements and refinement farm, compared for different size simulations and different numbers of PEs. The plot shows the time spent in each part of the calculation scaled to unity to allow comparison. Because the large level 1 refinements become very inefficient (maximum speed-up of 3) as the number of PEs is increased, the relative time increases significantly. Circumventing this problem is difficult.

Chapter 5

Galaxy formation in hierarchical clustering

"It will only be a few years before the origin and evolution of galaxies is understood."

–Yakov Borisovich Zel'dovich, 1977

5.1 Introduction

The previous chapters have shown that detailed studies of galaxy formation require many more particles than are currently used. This chapter details a simulation with sufficient resolution to overcome the criticisms presented. Because of the sheer computational effort required, the simulation uses the parallel code developed in chapter 4.

Much has been made in the literature about the 'overmerging' problem (Moore *et al.*, 1996; Moore *et al.*, 1998) in high resolution dark matter simulations. In a hierarchical clustering model, high density halos are formed by the accumulation of smaller ones. Simulations show that during this process, the substructure of the smaller halos is erased which is termed overmerging. While in the long term this is to be expected, since eventually halos will orbit close to the dense core and become tidally disrupted, the problem appears much quicker in simulations than is predicted on analytic grounds (Klypin *et al.*, 1999). Although the solution to the problem is to add more resolution, this is not always a satisfactory one. Most of the studies of this problem are performed in the galaxy cluster regime where the concept of 'galaxy harassment' is important. The simulation presented in this chapter is perhaps the first to exhibit this problem from the standpoint of galaxy formation, as a large number of halos merge to form the object of interest. It also includes hydrodynamics while none of the previous high resolution studies have. Thus, it is interesting to examine what happens when gas cores are added to the halos.

To date, the highest resolution studies of galaxy formation starting from realistic initial conditions are those of Navarro and Steinmetz (1997) which used up to 5,000 gas particles per galaxy. These simulations were integrated to $z = 0$ which resulted in the particles in the shortest time-step bin taking over 50,000 time-steps. Although the particle number is comparatively low, at least compared to that required for accuracy (*c.f.* the results for the velocity field of the spherical collapse, presented in chapter 2), the integration to $z = 0$ is a significant achievement. The purpose of the investigation was to study the evolution of systems with and without a photo-ionization background to see whether it suppressed disc formation. To ease analysis, the effects of star formation and feedback were neglected, and it was concluded that photoionization alone does not provide sufficient heating to prevent the formation of dense discs with an angular momentum deficit. Steinmetz and Muller (1994; 1995) considered rotating cloud models with star formation and comparable resolution to the Navarro and Steinmetz work. However, the initial conditions were idealized and no contribution from long-range tidal forces was included. Vedel *et al.* (1994) considered a similar model that includes the effect of UV flux on the gas cooling rate.

As indicated, results are reported in this chapter from a simulation with far higher mass resolution than previous models. Star formation and feedback are also included in the simulation using the energy smoothing model described in chapter 3. Since the simulation probes smaller mass scales than the low resolution work, predictions can be made about the effect of feedback in this regime. Consequently, the morphological results for objects in this simulation are of great interest. In chapter 3, concerns about the effect of increasing the resolution were expressed since feedback events

produce the same temperature distribution regardless of the underlying particle mass. This is a direct result of specifying feedback events using a specific energy rather than an absolute value. For lower mass halos, with consequently lower escape velocities, feedback regions should expel more gas than high mass systems. This was observed in chapter 3, where the NGC 6503 dwarf galaxy prototype was more affected by feedback than the Milky Way model.

It is important to re-emphasize how difficult it is to form a disc galaxy from the hierarchical initial conditions. The core-halo angular momentum transport mechanism (Barnes, 1992), is extremely pervasive in CDM cosmologies since galaxy halos grow by the accretion of smaller halos. Consequently, the fundamental tenet of the disc formation model in Fall and Efstathiou (1980), *i.e.* that the gas maintains a similar specific angular momentum to the dark matter during collapse, does not appear to hold for a CDM model. Results from Weil *et al.* (1998) have shown that the problem can be avoided if the underlying gas distribution is prevented from cooling until very late in the evolution of the cosmology ($z = 1$). Whether feedback can produce a similar distribution, and hence avoid the angular momentum problem, was examined in chapter 3. It was discovered that by using feedback it is difficult, but not impossible, to increase the ratio of the angular momentum in the gas cores to that of the dark matter halo. A small—but clear—trend towards an increased angular momentum for the gas core was observed with increasing feedback strength. The higher resolution of the simulation presented in this chapter will make the core-halo transport problem worse, if feedback is unable to keep the gas in a diffuse state.

The layout of the chapter is as follows: firstly the method used to find a suitable halo is reviewed. Since this section draws heavily from chapter 3, detail is kept to a minimum. Next the star formation model and numerical method are discussed. Parallelization of the star formation code was not discussed in chapter 4 and hence is reviewed here for completeness. Following this, results are presented, along with a short account of problems encountered during simulation. Next, a review of the SFR, morphology and the effect of feedback is presented. The chapter concludes with a discussion of the predictions of the simulation and its relation to observations of Lyman-break galaxies.

5.2 Initial conditions

Work presented in chapter 3 detailed the method for simulation of galaxy formation, including large-scale tidal fields. To increase the resolution of the simulation, more particles are necessary, and to study such problems within a reasonable time frame requires the use of parallel computing.

5.2.1 Low resolution dark matter simulation

The Bond and Efstathiou (1984) power spectrum with a normalization of $\sigma_8 = 0.6$ was used, which is the same as chapter 3. However, the box size was decreased, very marginally, to 48 Mpc. As a result of this decrease the high resolution region was 6 Mpc in diameter (compared to 6.25 Mpc in the low resolution studies). The 4% reduction in the box size leads to a 13% increase in the mass resolution and only a marginal loss of long-range tidal forces. The resolution of the lowest level grid was set at 100^3 and it was decided to use this particle number in the collisionless simulation. Thus the mass resolution per particle was 7.7×10^9 $M_\odot$ so that a 10^{12} $M_\odot$ halo is resolved by 130 particles. This resolution is sufficient to be able to pick out halos of this size, but is not sufficient to make detailed predictions about their internal structure. An initial redshift of $z = 67$ was chosen since early stages of the evolution can be calculated efficiently and choosing a higher z reduces the effect of errors in the initial application of modes calculated via the Zel'dovich approximation.

Since 100^3 is not a power of 2, a new glass had to be constructed. The same procedure as in chapter 3 was followed, except that the 'tile' was constructed from 1000 particles. A plot of the power spectrum for the 100^3 grid created by replicating the tile is shown in Figure 5.1. The noise level (measured by the bin averages) is at least four orders of magnitude lower than Poisson noise at any scale, and three orders of magnitude lower than the applied perturbation spectrum. Because the tile is replicated ten times, there is a significant spike in the spectrum at k=10. However, the mean of the bin is still three orders of magnitude lower than the applied perturbation spectrum.

5.2.2 Selection of resimulation halo

The collisionless simulation was evolved through 1145 time-steps to $z = 1$ using a softening length of 16 kpc. At this epoch a 'friends-of-friends' group finder was applied to find all halos of mass greater than 1.2×10^{12} $M_\odot$ (those containing more than 156 particles). The linking length was set to 20 kpc (only 25% larger than the softening length), which at $z = 1$ is 8.3×10^{-4} times the box size. The group finder returned 261 halos satisfying this criterion, the maximum mass being 1.2×10^{14} which

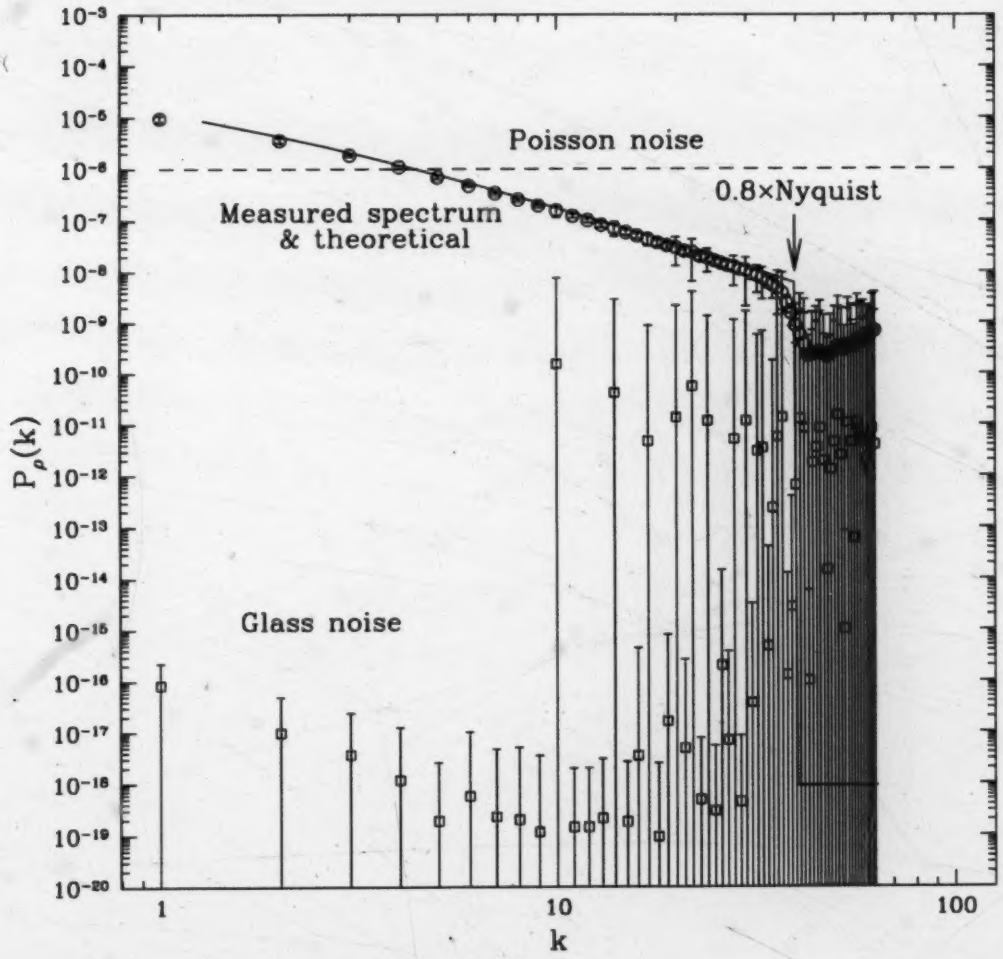

Figure 5.1: Power spectra for the glass noise, Poisson noise and the applied perturbation spectrum. The glass noise level (measured by bin averages) is five orders of magnitude below the Poisson noise and even 1σ deviations are still at least two orders of magnitude lower. The measured spectrum is very slightly lower (5%) than the applied one, but since statistics are not being measured, this is not a cause for concern. The cut-off in the applied signal comes from a sharp exponential filter which is applied at 0.8 times the Nyquist frequency. This filter prevents sudden growth on short scales which can significantly affect the mode coupling within the box once the simulation becomes non-linear.

Table 5.1: Summary of simulation parameters

Parameter	Value
Full comoving box size (Mpc)	48
High resolution region (Mpc)	6
Mass in HR region ($M_\odot$)	8.0×10^{12}
Initial number of particles in HR region	$2 \times 523,535$
Effective resolution of simulation	2×800^3
Gas particle mass ($M_\odot$)	1.5×10^6
Minimum glob resolution ($M_\odot$)	8.3×10^7
Dark matter particle mass ($M_\odot$)	1.4×10^7
Number of star particles at final epoch	40,777
Minimum dark halo resolution ($M_\odot$)	7.4×10^8
Gravitational S2 softening, (kpc)	3.51
Equivalent Plummer softening, ϵ (kpc)	1.5
Minimum hydrodynamic scale, h_{min} (kpc)	1.76
Initial temperature, (K)	1,000
Star formation rate normalization, $c*$	0.015
Feedback energy (erg g^{-1})	5×10^{15}
Half-life of feedback region (Myr)	5
Initial redshift, z	67
Final redshift, z	2.16
Average time-step (Myr)	0.56
Final time-step (Myr)	0.32
Total number of iterations	4,100
Predicted number of iterations to $z = 0$	37,000
r_{200} at $z = 2.16$ (kpc)	75

corresponds to the largest galaxy cluster in the volume. The five halos closest to 1.5×10^{12} $M_\odot$ in mass were then examined to determine their environments. A field galaxy, *i.e.* one distinct from clusters, is ideal for simulation since the effects of movement through a gas medium are less relevant. In multiple mass simulations, this is necessary since the high resolution gas region does not extend out more than 3 Mpc. Consequently, of the five halos examined, the chosen one corresponded to a halo resident on a filament which was not too close to any clusters. The exact mass of the halo was 1.66×10^{12} $M_\odot$, and it was resolved by 217 particles. At this resolution, it is impossible to detect any internal structure. A zoom-in on the candidate halo is shown in Figure 5.2 and it displays both the global and local structure in the simulation.

Once the particles in the halo had been found, they were tagged, thereby allowing the initial positions to be determined. The center of mass of the system in the initial conditions was evaluated and then the nesting of the grids was centered on these coordinates. The radius of the 3 Mpc high resolution sphere was compared to the initial positions of the dark matter particles resident in the chosen halo to assure that it completely encompassed them. This was indeed the case, and as in chapter 3, a small boundary, approximately 10% of the width of the high resolution region, exists between the sphere and the halo particles. The 100^3 glass was nested in the same fashion as chapter 3. A total of $2 \times 523,535$ particles were placed in the 6 Mpc diameter high resolution region, half were gas particles, half were dark matter. The modes of the initial power spectrum were represented on a 1024^3 grid, which was filtered at the Nyquist frequency corresponding to the average inter-particle spacing of each hierarchy. The effective resolution of the highest resolution region is 2×800^3, *i.e.* it is equivalent to a fixed resolution simulation with a little over one billion particles. Within this region the gas particle mass is 1.5×10^6 $M_\odot$, the star mass 7.5×10^5 $M_\odot$ and the dark matter resolution is 1.4×10^7 $M_\odot$. Consequently at $z = 1$ the resimulated halo is represented by 110,000 dark matter particles and an equal number of gas particles.

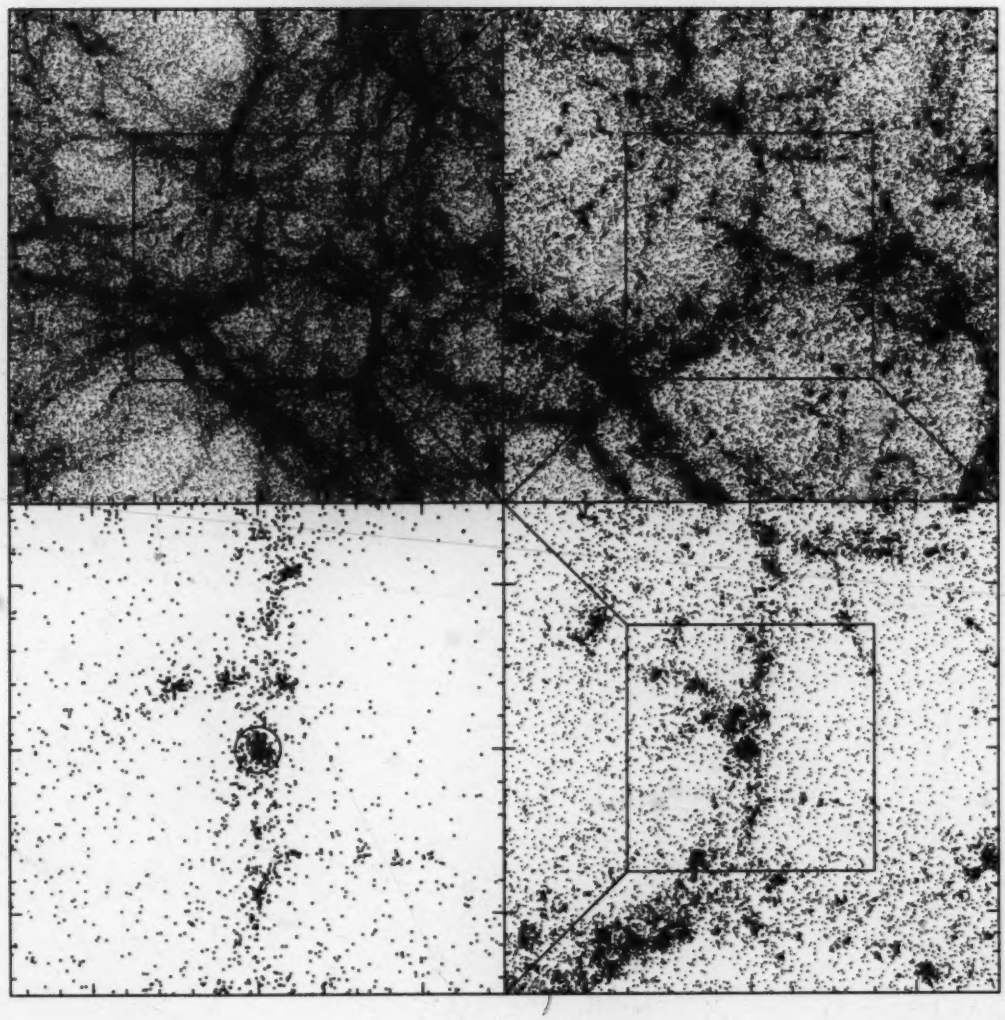

Figure 5.2: Zoom in on the resimulation halo in the 100^3 simulation to show the local and global environment. Starting at the top left panel and moving clockwise, the first panel is 24 Mpc wide, while the final is one-eighth the size, *i.e.* 3 Mpc across. The candidate halo sits in the middle of a filament approximately 2 Mpc long.

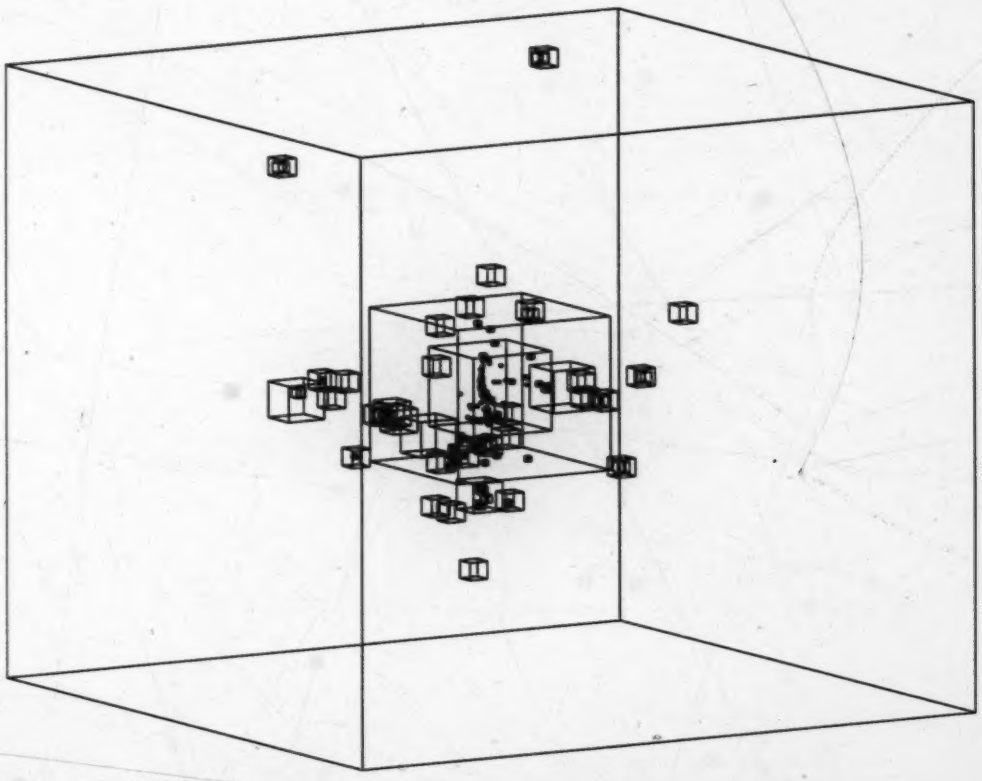

Figure 5.3: Placement of the adaptive refinements within the entire simulation box at $z = 2.2$. The two large central refinements correspond to the nested 256^3 refinements which are placed by hand. There are comparatively few refinements in the outer-lying regions since the ratio of the particle density to grid spacing is small and also the simulation has only evolved to $\simeq 20\%$ of t_0.

5.3 Simulation algorithm

The results from chapter 3 showed that the temperature smoothing algorithm produced the most significant feedback effect. However, it was noted that smaller mass scales will be more affected by feedback since the ratio of the binding energy to feedback energy is smaller. Thus, increasing the mass resolution should make the energy smoothing algorithms more efficient. Hence, for this simulation, the ESa algorithm was adopted with the energy return value set to $e^* = 1$ and the feedback region half-life, $t_{1/2}$=5 Myr. Since higher densities are resolved by the simulation, it was decided to slightly reduce the SFR normalization, $c^* = 0.015$, compared with that in the isolated simulations, namely $c^* = 0.025$. The self-gravity criterion was kept at 0.4 times the dark matter density.

5.3.1 Parallelization of the star formation algorithm

Parallelization of the star formation algorithm was comparatively straightforward. Since only very few particles are converted from gas to star particles during a step it is possible to use locking on the particle data without any contention occuring. For ease of programming, a linked list of the *gas particles* in the simulation was constructed using a 128^3 chaining cell grid, spaced over the extent of the gas region. Although it has been demonstrated that ordered lists or ordered particles are

more efficient, it was decided to keep the programming of the star formation algorithm as simple as possible since it does not constitute a large fraction of the run time (10%). The grid was allowed to increase and decrease in size as the maximal extent of the high resolution gas region changed. Evaluation of the star formation occurs by sorting through the particles in each cell. Initially a 1-d slab decomposition of the cells was used, but this quickly proved to be inefficient (the execution time of the routine was found to be very sensitive to clustering). This problem has been seen before (see chapter 4) and was solved by the use of a 2-d block decomposition. Hence, the routine was reprogrammed to use a dynamic distribution of columns which, because of the locking mechanisms, do not need any *a priori* ordering. The load balance could also have been improved by estimating the load in each block, but this was not done.

5.3.2 Refinement placing

Placement of the adaptive refinements was controlled in a similar fashion to that in chapter 3. However, to reduce the placement of $L = 64$ refinements which are particularly inefficient, a series of $L = 256$ refinements were nested. The first 256^3 refinement was placed over the region enclosed by the second highest hierarchy while the second one covered the high resolution region itself. Although placing the second 256^3 refinement leads to some inefficiency, the trade-off is beneficial as the more inefficient 64^3 that would have been placed are removed. Figure 5.3 shows the placement of the adaptive refinements within the entire simulation box. In the high resolution region, the refinements are clearly placed along the vertical filament which the halo forms along (compare to the final panel of Figure 5.2).

5.3.3 Inefficiencies in the algorithm

In the unclustered state, the algorithm took only 30 seconds per time-step on 32 nodes of an SGI Origin 2000 system. However, as clustering developed, severe inefficiencies began to show up in the algorithm, and by $z = 2.16$, the execution time had risen to 1,900 seconds which is a more than 60-fold increase. This is both a result of the larger amount of work developed as many particles reach the h_{min} smoothing limit, and a load imbalance caused by certain chaining cells taking far longer to calculate than others. These inefficiencies are discussed in chapter 4. The slow-down in wall-clock performance prevented the simulation from being evolved to the desired final epoch of $z = 1$. Instead, the simulation was stopped at $z = 2.16$, by which point, the simulation had evolved through 4,100 steps. Integrating beyond this point, using the current algorithm, is simply not feasible. For a wall-clock time of 2,000 seconds per time-step, another 10,000 steps would require *230 days* of computing time. An improvement to the algorithm is clearly warranted, as was suggested in chapter 4. It should be emphasized that the problem is not intractable; for example, the dark matter calculation was shown to only require 130 seconds in the clustered state. However, it does require a careful re-examination of the SPH solver and how it is combined with the dark matter solver. Assuming that the final time-step value, δt, is a reasonable average for the remainder of the simulation, this value suggests the total number of time-steps to $z = 0$ is about 37,000. Bearing in mind that at later epochs higher in-fall velocities result due to higher densities, the total number of time-steps would probably be closer to 50,000. Because of the truncated evolution, the subsequent analysis has been altered to examine the details of the progenitor halos, and the very earliest stages of evolution of the central galaxy. Some data was lost when the auto-eraser on the UK-CCC supercomputer erased some data logs which exceed the seven day age-limit. Fortunately, the main simulation file was not lost. The missing data were not recomputed since they only constitute a very small fraction of the entire data set (less than 5%).

5.4 Results

A summary of the simulation parameters is given in table 5.1. A number of results for the final state at $z = 2.16$ are also given.

The structure and evolution of the gas in the simulation is depicted in the series of figures 5.4-5.7. The structure resolved in the high resolution has a mass resolution over ten times higher than any previous simulation of a similar volume. In much the same way that the simulations by Moore *et al.* (1998) show how halo substructure contributes to the formation of galaxy clusters, the simulation presented here shows how substructure contributes to the formation of galaxies. Note that the selection of a conservative softening parameter means that the force is not resolved to as small a fraction of the virial radius, r_{200}, as the simulations of Moore *et al.*. For the simulation presented, the force is resolved to 2% of the virial radius at $z = 2.16$ whereas Moore *et al.* present simulations resolved to well under 1%.

In figures 5.4 and 5.5, the evolution of the comoving density and physical temperature are displayed using comoving coordinates. The images were constructed as follows: The SPH data were interpolated on to a grid with spacing set by h_{min}, using a 2-dimensional Gaussian smoothing kernel. The width of the kernel was set by $r_0 = 0.585h$ to approximate the width of the B2-spline used in the SPH calculation. The 2-dimensional projection is quite accurate since the integration of the 3-dimensional Gaussian kernel in the projection direction is a summation of 2-dimensional kernels multiplied by the error function. Further, the error function only has a significant contribution at the edge of the interpolation, and hence can be ignored since the kernel decreases rapidly in this region.

Comoving coordinates have been used since there is a factor of 3.5 difference between the physical size of the data in panel at $z = 10$ and that at $z = 2.16$. Consequently, the average physical density is 40 times higher. By $z = 10$, five gas cores are visible (see the plot of identified groups in Figure 5.13) and they range in size between 10^8 and 6×10^8 $M_\odot$. The dark halos in which they reside are approximately nine times more massive (*i.e.* the ratio Ω_{dm}/Ω_b), and vary between 10^9 and 5×10^8 $M_\odot$. Filamentary structure, 'the cosmic web' (Bond *et al.*, 1996), is resolved at this very early stage of the evolution. A small fraction of the gas has been shocked to temperatures greater than 5,000 K, but otherwise there has been very little thermal evolution. The majority of the gas simply cools due to the PdV work done in the adiabatic expansion (the gas is below the 10,000 K limit of the radiative cooling curve).

By $z = 5$, structure formation has advanced considerably, and well over 100 dark matter halos and globs are resolved. The largest dark matter halo at this epoch has a mass of 5×10^{10} $M_\odot$ and the glob associated with it has a mass of 5×10^9. At this stage of evolution, the largest globs overcome the self-gravity criterion and begin to form stars (see Figure 5.15 for a plot of the SFR *vs.* time and section 5.4.4 for a discussion of it). The filamentary structure is more pronounced than at $z = 10$. More shocking has occurred in the gas, and there is a visible build-up of heated gas around the bottom of the potential well that will accommodate the galaxy.

Between $z = 3$ and $z = 5$, a large fraction of halos merge, forming more massive structures. The filamentary structure, seen in the density plot, is even more pronounced at $z = 3$ than at $z = 5$, and it is clear that a large fraction of in-fall to the central object occurs along a filament which the z-projection projects directly along (see Figure 5.6 for three projections of the $z = 2.16$ density and temperature distributions). The temperature plot shows the early development of the hot gas halo associated with the dark matter potential. A few particles at the bottom of the potential are heated to over 10^6 K.

By $z = 2.16$, the hot halo has evolved further and the hottest particles reach 10^7 K. However, there are fewer than ten particles above this value and the radially averaged temperature profile (in bins of size 208 particles, see Figure 5.19) suggests the average central temperature is closer to 10^6 K. Two other hot halos are beginning to form in the upper region of the plot, although these regions will not merge until after $z = 1$. The small lines in the temperature plots are associated with gas that is shocked as it falls on to a filament (most of this gas is at 10,000 K). This effect is visible at earlier redshifts. The density plot shows that the distribution continues to evolve from $z = 3$ to $z = 2.16$ and the hierarchical merger process continues to form halos of larger masses. At this epoch, the largest dark matter halo has a mass of 6×10^{11} $M_\odot$, while the largest glob has a mass of 6×10^{10} $M_\odot$.

The comparison of temperature and density shown in Figure 5.6 gives a clear indication of the projection effect discussed earlier. When viewed along the z-axis (top panel) the halo appears to be quite clustered. The y- and x-projections, in the middle and bottom panels respectively, show that the matter distribution is still quite extended, and collapse into the main halo is occurring largely along the dominant central filament. The projections show that the hot halo is approximately spherical. This is to be expected since the sound crossing time is

$$t_{cross} = 6.6 \times 10^7 \left(\frac{T_{gas}}{10^6\,\mathrm{K}}\right) \left(\frac{D}{\mathrm{Mpc}}\right) \mathrm{yr}, \tag{5.1}$$

where $T_{gas} \simeq 10^6$, and the width of the halo is $D = 0.5$ Mpc yielding $t_{cross} \simeq 0.03$ Gyr. Compared to other filaments in the simulation, the build-up of shocked gas around the central filament is much larger.

The final plot in the series shows a zoom-in on the main halo. The gas density is shown and the the colour scheme spans three orders of magnitude. Since each panel is a projection of a box of depth equal to the width, material lying behind the smaller panels is lost. Hence, the smaller panels accurately depict local structure. The series of plots also graphically demonstrates the large dynamic range in the high resolution region. In the final plot, 8 halos can be identified that are at the maximum density limit, and a further 20 with lower masses are visible. Since most of the halos are only 50 kpc distant from the core, they can be expected to merge with it before $z = 1$.

These plots demonstrate beautifully that smooth in-fall is *not* the main method for baryon accretion in CDM models, contrary to the results from very low resolution simulations where the halos are simply not resolved.

5.4.1 Morphology and the effect of feedback

The 1.5 kpc resolution allows for density values that are 20 times higher than the low resolution simulations, which were limited a the 4 kpc softening length (as discussed in chapter 3). Note that the size of the S2 softening relative to the grid spacing is 0.15 for the high resolution simulation, compared to 0.05 for the low resolution simulation. Thus, the high resolution simulation is more conservative in this regard and suffers less from contamination due to two-body relaxation. Since in the hierarchical clustering picture the smallest halos form first, increasing the resolution leads to more halos being resolved at earlier times. Consequently, the gas overcomes the self-gravity criterion, and hence forms stars, at an earlier epoch ($z = 5$) than the lower resolution simulations ($z = 4$).

The disc-like objects formed in this simulation do have a different structure to the major disc examined in chapter 3. In the low resolution disc, a very dense gas core, with a spatial extent smaller than $0.1h_{min}$, was formed in nearly all simulations, the exception being the temperature smoothing feedback. In Figure 5.8, a projection of the gas and star particles in the two main dwarf systems, denoted DW1 and DW2, is shown at a redshift of $z = 2.2$. In total, the baryons in DW1 are represented by 18,000 particles, while DW2 is represented by 30,000 particles. DW1 does *not* have a dense gas core similar to the low resolution disc and, although DW2 has a dense region within the 'ring', it is more than 1 kpc in diameter. Unfortunately, the spatial extent of these systems is only 4 kpc, and hence, given that $h_{min} \simeq 1.5$ kpc, they are very poorly resolved. For example, the 'ring' feature in DW2 is only 1 kpc wide and hence is not really resolved at all. Visualization of the dwarfs shows that feedback causes small pockets of hot gas which remain static until the region reaches the set half-life at which point the gas cools rapidly.

Because of the high mass resolution in the simulation, tidal 'tails', formed during the interaction of halos (see Toomre and Toomre, 1972, for seminal work on these features), are resolved extremely well. In Figure 5.9, the gas particles involved in an interaction between two dwarf systems are shown. Note the larger system is the DW1 dwarf. Gas is stripped from the edge of DW1 and forms a bow-wave around the smaller system. The encounter occurs at $z = 2.6$, at which time, the stellar content of DW1 is still small, 3×10^9 $M_\odot$, and hence, no stellar matter is stripped. Also note that the viscosity of gas tends to accentuate the appearance of stripped matter. Collisionless matter, namely dark matter and stars, is less susceptible to this effect. These interactions are of particular interest in hierarchical cosmology, especially in galaxy clusters, since they predict that there should be a small fraction of intra-cluster light due to the presence of stellar matter in the tails (Gregg and West, 1998).

As the simulation evolves, tails become more prevalent. An examination of the final state, shown in Figure 5.17, shows that the bulk of the gas objects merging with the central core are tidally stripped as they merge. This explains the origin of the ring feature of DW2, it was formed by a system becoming tidally stretched as it fell on to DW2. This feature is not formed by the same mechanism as nuclear rings in galaxies, which are a result of torques on the disc matter from the bar instability (Schwarz, 1981, 1984).

5.4.2 Angular momenta of the dwarfs and main halo

To analyze the growth angular momentum in the system, the same procedure was adopted as in chapter 3. In Figure 5.10, the z-component of the angular momentum in the DW1 and DW2 dwarfs at $z = 2.2$, is compared to the Rv_c prediction. The two dwarfs have well-defined discs, albeit with a comparatively low aspect ratio since the disc thickness is about 1 kpc, while the diameter is about 4 kpc (which is smaller than $4h_{min}$). Compared to previous studies, *e.g.* Navarro and Steinmetz (1997), both DW1 and DW2 have good mass resolution, since they have at least double the number of gas particles of the largest objects formed in the older simulations. For DW1, most of the gas and star particles have L_z values slightly under the Rv_c prediction, but do seem to follow the shape of the predicted curve reasonably well. There is a slight downturn in the slope at the edge of the disc, the origin of this feature is not understood. Since a large fraction of the L_z values lie comparatively close to the predicted curve, DW1 has not yet lost a significant amount of angular momentum due to bar formation. Given that the structure of the disc is sub-resolution, this would not be expected. According to the $X_2(R)$ data, the disc achieves stability at a radius slightly under 2 kpc, which is at the very edge of the disc. The results for DW2, the more massive of the two systems, are comparatively similar. The bulk of the L_z values again lie close to the Rv_c prediction. At larger radii, there appears to be a considerable fraction of mass, at least compared to the DW1 result. For this matter, the L_z values do appear to lie under the Rv_c prediction, suggesting that angular momentum has already been lost. The core-halo mechanism has probably had an effect on this

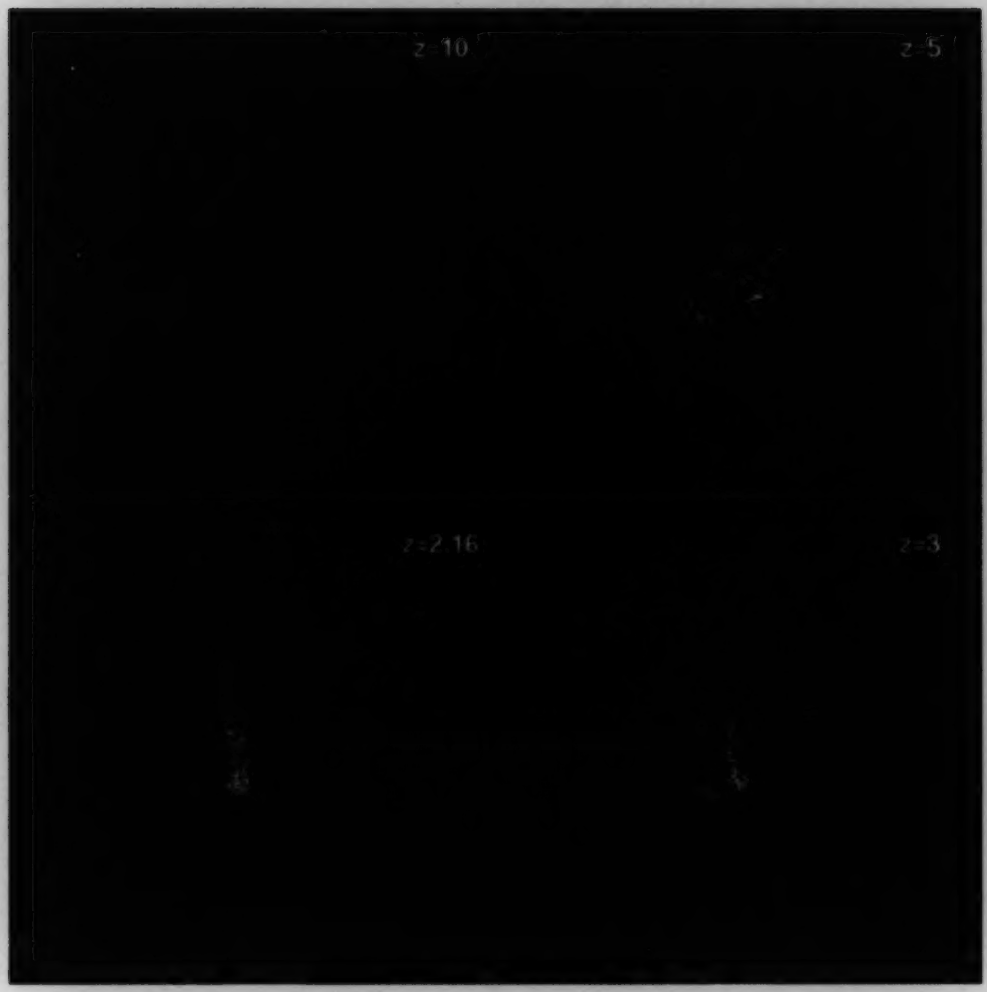

Figure 5.4: 4-panel plot showing the evolution of the comoving density from $z = 10$ to the final epoch $z = 2.16$. The SPH data is smoothed onto a grid with spacing h_{min}, thus the grids are a realistic representation of the resolution. In physical coordinates the top left panel would be 3.48 times smaller than the bottom left. The colour scheme runs from 10^{18} n_B cm^{-2} (blue) to 10^{21} n_B cm^{-2} (red). The filamentary structure is already forming at $z = 10$ and by $z = 5$ the first halos have reached sufficient density to form stars. Evolution from $z = 3$ to $z = 2.16$ is dominated by collapse along the x-direction. Note that the z-projection looks directly along a filament and thus over-emphasizes the collapse. See Figure 5.6 for x-, y- and z-projections of the density.

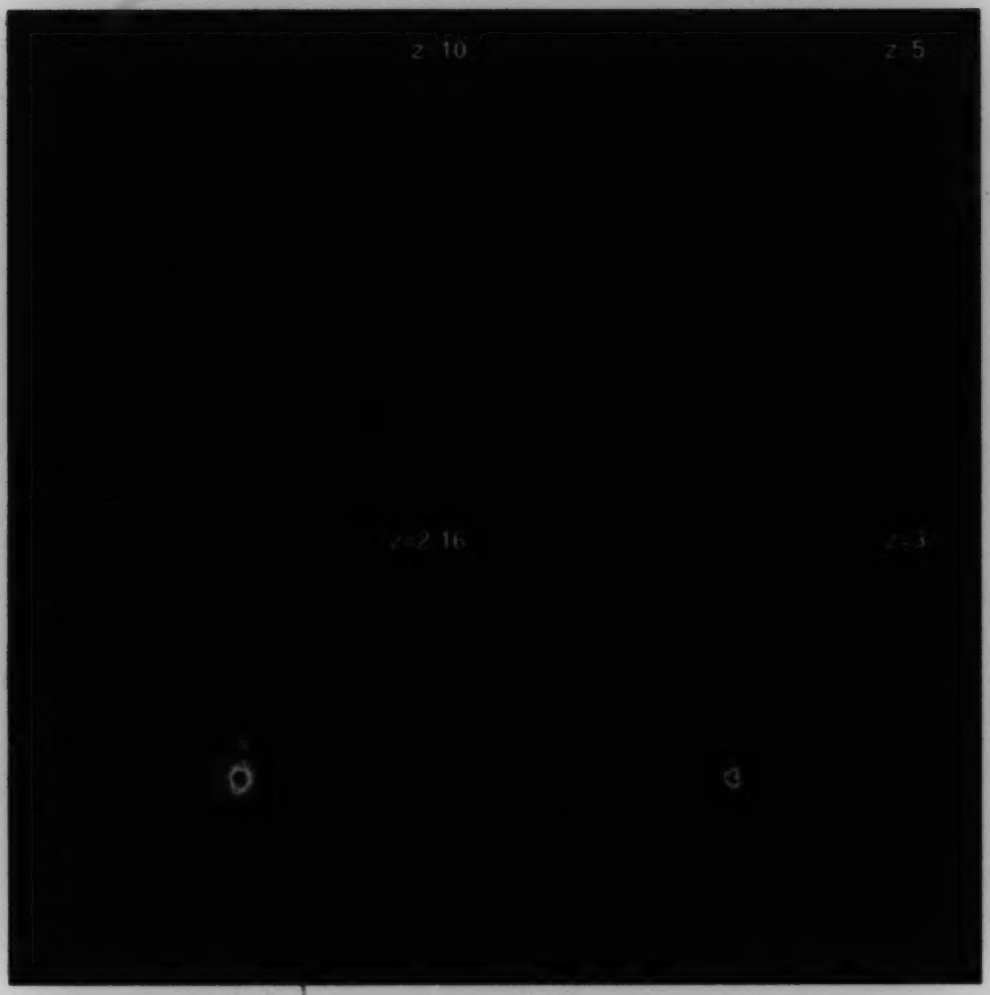

Figure 5.5: 4-panel plot showing the evolution of the physical temperature from $z = 10$ to the final epoch $z = 2.16$. The SPH data is smoothed using the same procedure as in Figure 5.4. The spatial coordinates are comoving so that the top left panel should be 3.48 times smaller than the bottom left. The colour scheme for the physical temperature runs from 5×10^3 K (blue) to 10^7 K (red). The hot gas halo does not grow significantly until $z = 5$, since it is dependent upon the collapse of the dark matter potential well.

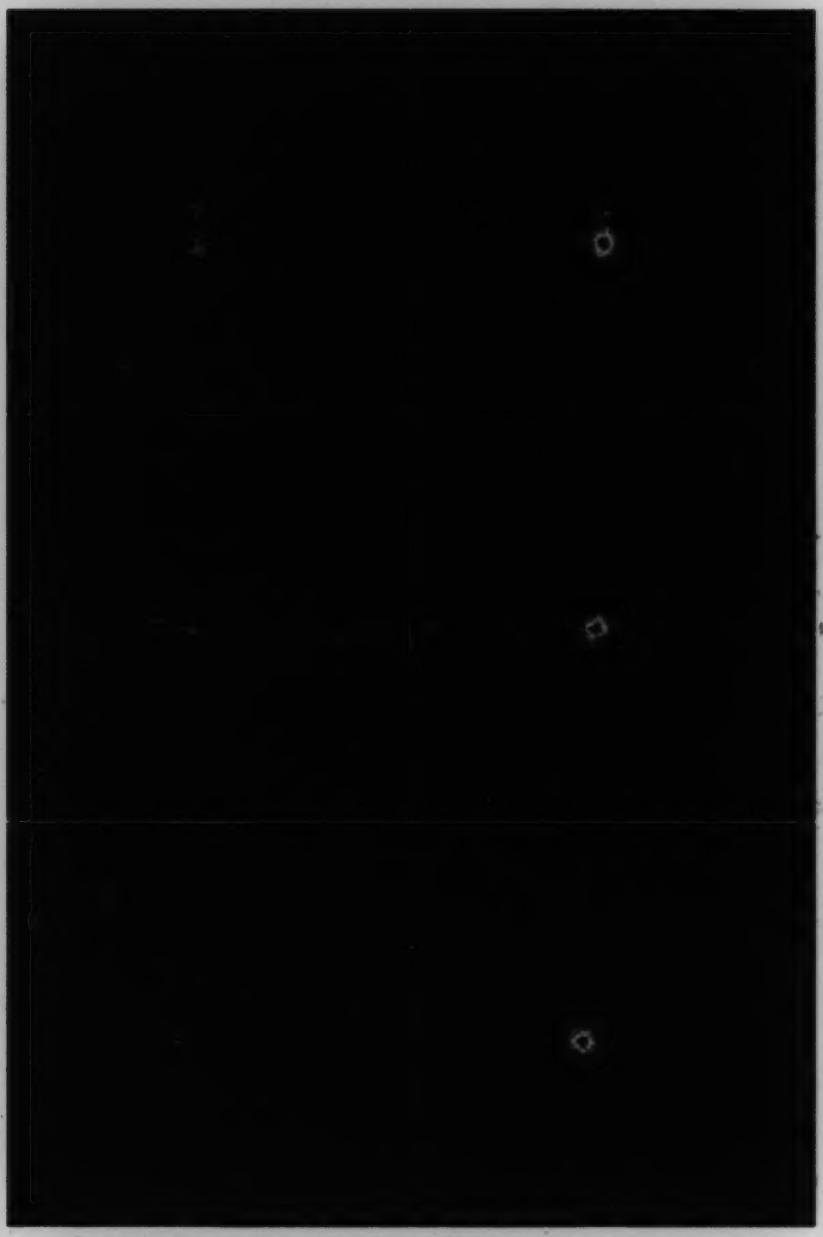

Figure 5.6: Projections of the gas temperature and density at $z = 2.16$. The density is given in the left-hand panel while the temperature is in the right-hand panel. From top to bottom are the z-, y- and x-projections. Note that the y- and x-projections are not taken from rotations of the z-projection. Each panel is 1.89 Mpc across, and all values are given in physical units. The density colour scheme runs from $3.2 \times 10^{19}\ n_B\ \mathrm{cm}^{-2}$ (blue) to $3.2 \times 10^{22}\ n_B\ \mathrm{cm}^{-2}$ (red). The temperature colour scheme runs from 3.0×10^3 K (blue) to 1.0×10^7 K (red).

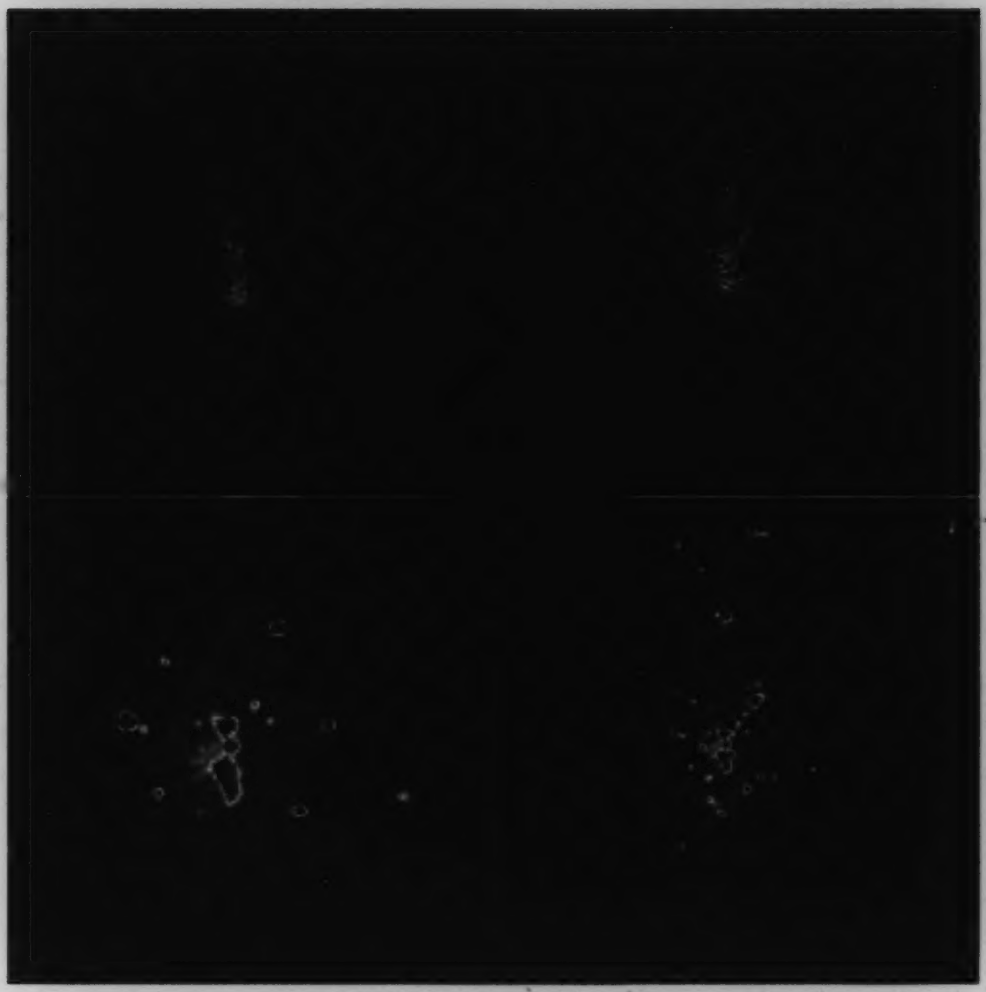

Figure 5.7: 4-panel zoom in on the gas in the high resolution region to show environment. The first panel is 1.89 Mpc wide and each succeeding panel is half the size of the previous (in clockwise order). The density colour scheme runs from $3.2 \times 10^{19}\ n_B\ \mathrm{cm}^{-2}$ (blue) to $3.2 \times 10^{22}\ n_B\ \mathrm{cm}^{-2}$ (red). The final panel is just over 236 kpc wide and shows the clustering of the halos that are merging to form the galaxy. There are 138 cells of size h_{min} in the final panel, indicating that it is still well resolved.

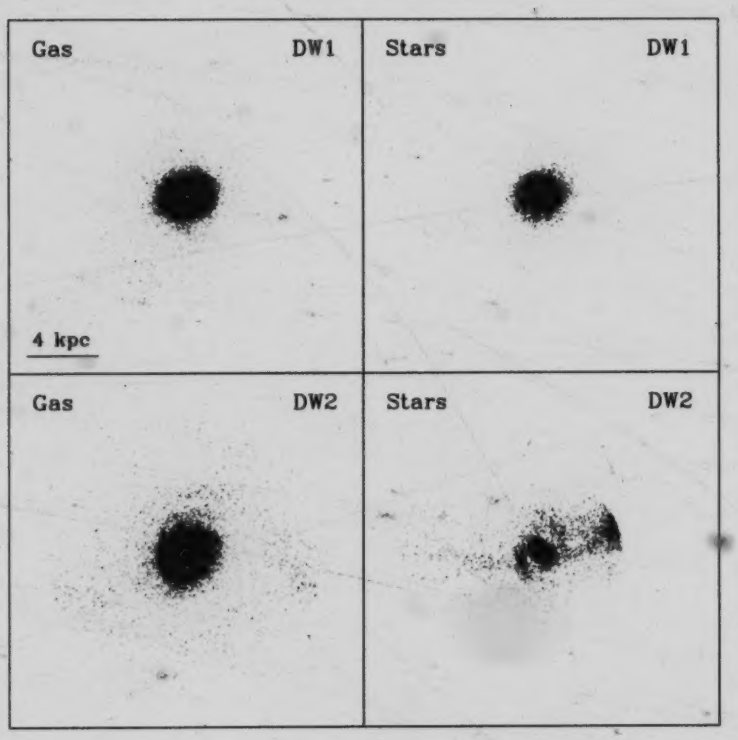

Figure 5.8: Projections of the gas and star particles in the two largest dwarf systems at $z = 2.2$. The particle positions are projected from spheres of radius 8 kpc. DW1 has a gas mass of 1.0×10^{10} $M_\odot$, which is represented by over 10,000 particles while the stellar mass is 5.8×10^9 $M_\odot$, corresponding to 7,800 particles. DW2 has a gas mass of 2.5×10^{10} $M_\odot$, represented by almost 18,000 particles and the stellar mass is 8.9×10^9 $M_\odot$ corresponding to almost 12,000 particles. DW2 is more massive and is the site of most of the accretion. At $z = 2.16$, DW1 collides with DW2. Note that the star morphology of DW2 shows a 'shell' of star particles due to a recent merger. This feature is observed about a number of elliptical galaxies (see Quinn, 1984, for a discussion of the formation of shells). The gas particles in the merger have wrapped around the central gas nucleus to form a visible ring. Since $h_min \simeq 1.5$ kpc both of these systems have very poor linear resolution.

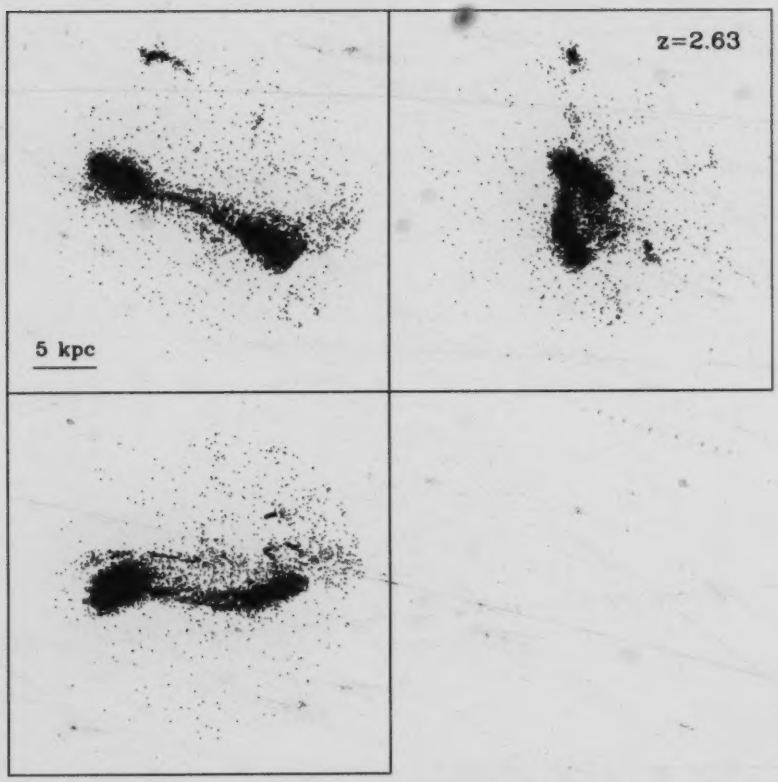

Figure 5.9: Three projections of a gaseous tidal tail formed by the interaction of two dwarf galaxies. The projections correspond to 90 degree rotations of the top-left panel. The high mass resolution of the simulations allows features like this to be observed in great detail. The more massive of the two dwarfs, is the DW1 system, and at this redshift it has a baryon mass of 1.2×10^{10} (8,000 particles) while the smaller system (unnamed) has a mass of 5.2×10^{9} (3,500 particles). The smaller dwarf has an in-fall speed of 200 km s^{-1} in the center of mass frame of the larger galaxy, and stripped matter from the larger dwarf forms a bow-wave in front of the smaller system after the interaction.

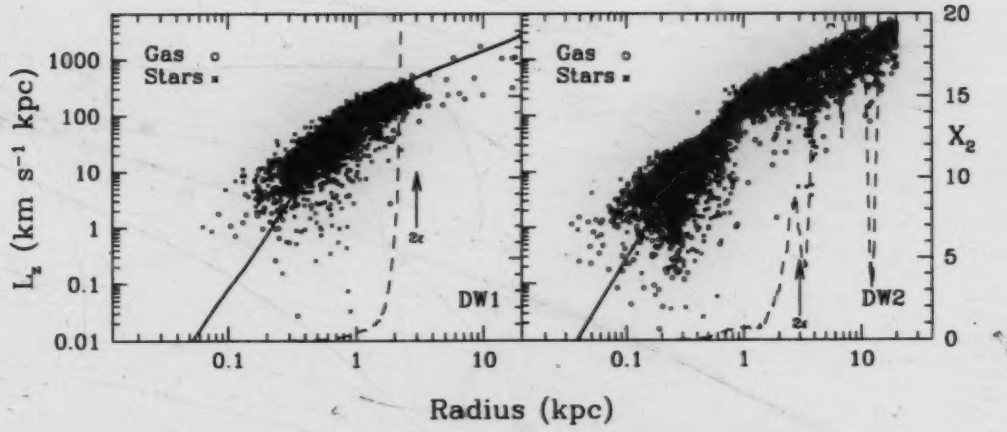

Figure 5.10: Plot of the z-component of the angular momentum of the dwarf systems DW1 and DW2, compared to the $L_z = Rv_c$ prediction. Most of the in-falling matter lies close to the prediction, indicating that these systems have not suffered great angular momentum loss. However the systems are very compact, with almost all the matter being enclosed within $2h_{min}$. Also shown (dotted line) is the $X_2(R)$ stability parameter for the dwarfs. The DW2 values are contaminated by particles just above the disc plane which are included in the surface density calculation but not shown in the particle plot.

merger. The $X_2(R)$ values show that DW2 achieves stability at a similar radius, namely 2 kpc, as DW1.

The same L_z analysis was applied to the baryon condensation in the main halo at $z = 2.16$. Although no clear disc is yet visible, visualization shows that a number of in-falling systems are in a orbiting in a similar plane. Provided that these systems contribute the largest fraction of orbital AM to the system, the dominant angular momentum component should be perpendicular to this plane. The results from this analysis are shown in Figure 5.11. No $X_2(R)$ plot is shown because no clear disc has formed. The data show that almost all the in-falling matter, picked out in the horizontal plane perpendicular to L_z, has lost a significant proportion of angular momentum relative to the Rv_c prediction. Since no disc has formed, all of this angular momentum loss must be due to the core-halo transport mechanism. This result appears to show that at higher resolution the angular momentum loss seems to be greater. However, caution should be emphasized in interpreting this result: if the selected matter just happens to be passing through the selected plane then its angular momentum vector does not align well with that of the entire system and hence the L_z analysis will overestimate the angular momentum loss.

The specific angular momenta for the dark matter, gas cores, gas halo and star particles were also calculated. Figure 5.12 displays the data for the main halo at $z = 2.16$, compared at r_{200} and $r_{200}/2$. The data suggests that the gas cores have not yet lost a significant amount of specific angular momenta. However, this would be expected since the largest part of the angular momentum is carried by the orbiting systems that are yet to merge with the central system. Only after these systems have merged, would it be expected that the ratio of the gas core angular momentum to the dark matter would be small. For the star particles, the r_{200} value is again similar to that for the dark matter. However, the $r_{200}/2$ value is much lower since an in-falling satellite, which carries the largest part of the angular momentum in the r_{200} value, no longer contributes. It is instructive to compare the results in chapter 3 to the $r_{200}/2$ ratio of the star particle angular momentum to dark matter at $z = 2.16$. For the low resolution simulations the ratio is in the range 0.10 to 0.15 at $z = 1$, while for the high resolution simulation the value is 0.19. Hence, the angular momentum loss in the core region is occuring earlier in the formation of this galaxy. This is presumably due to the higher resolution resolving more halos.

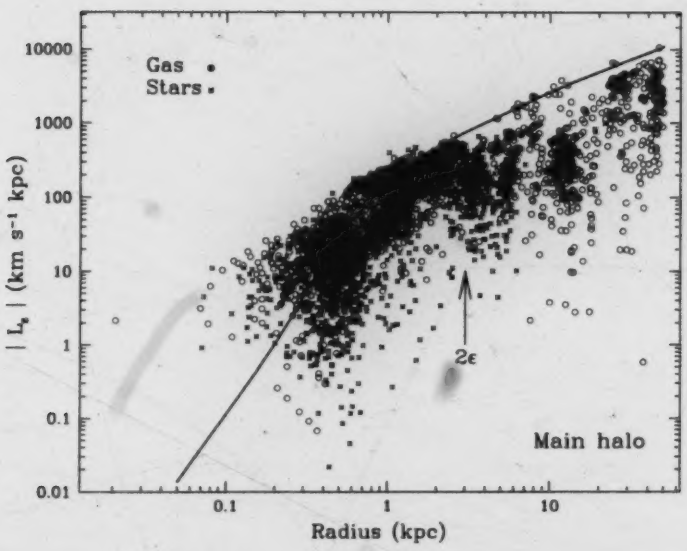

Figure 5.11: Plot of the z-component of the angular momentum of the main halo at $z = 2.16$ compared to the $L_z = Rv_c$ prediction. Contrary to the plots for the dwarf systems, a large fraction of the in-falling matter lies below the predicted curve. This is primarily due to the fact that the in-falling satellites are not on circular orbits, thus **r.v**, does not have a maximal angular part. The $X_2(R)$ values are not plotted since no disc has yet formed.

5.4.3 Halo and glob mass multiplicity functions

Three redshifts were selected for analysis, namely $z = 10, 5, 2.16$. To identify halos the 'friends of friends' (FOF) group-finding algorithm of Davis *et al.* (1985) was employed. The FOF group finder identifies halos for which all particles have an inter-particle spacing less than some value r, the 'linking length'. In terms of the original average inter-particle spacing, d, (for a 3-dimensional grid of size L, with N particles overlaid, $d = L/N^{1/3}$) the linking length for dark matter was $r_{DM} = 0.15d$. Since the baryon groups are more compact than the dark halos in which they reside, a shorter linking length was used to identify them. The prescription of Evrard *et al.* (1994) was adopted so that at $z = 10, 5, 2.16$ the linking lengths were $r_{bary} = 0.11d, 0.06d, 0.03d$ respectively. Note that at the final redshift, the groups found for the gas were less sensitive to the linking length than the dark matter. This is because the baryon groups have a sudden fall-off in density when compared to the dark matter. A z-projection of the groups found at the chosen epochs is shown in Figure 5.13.

In Figure 5.14 the cumulative mass multiplicity function for the dark-matter halos and globs is shown. The trend from left to right is due to halos collapsing and accreting mass. The data are fit well by an M^{-1} power law. This is shallower than the observed M^{-2} power law in Evrard *et al.* (1994). The difference is because the data presented here are biased: the measurements are made in a significantly overdense region which means that the halos must be biased toward heavier values. The tilt in the power law cannot be due to feedback blowing apart small baryon cores since both the dark matter and the baryons exhibit a similar slope. There is noticeably more evolution in the globs between $z = 5$ and $z = 2.16$ than there is for the dark matter halos. While the shapes of the two curves are similar at both epochs, the baryon curve shifts rightward at $z = 2.16$, indicating higher relative masses at this time.

5.4.4 Star formation rate

Although gas cores are beginning to form at $z = 10$, none of them overcomes the self-gravity criterion until $z = 5$. Hence the onset of star formation is delayed until this epoch, which is clearly visible in the plot of the SFR *vs.* time in Figure 5.15. The higher resolution in this simulation leads to

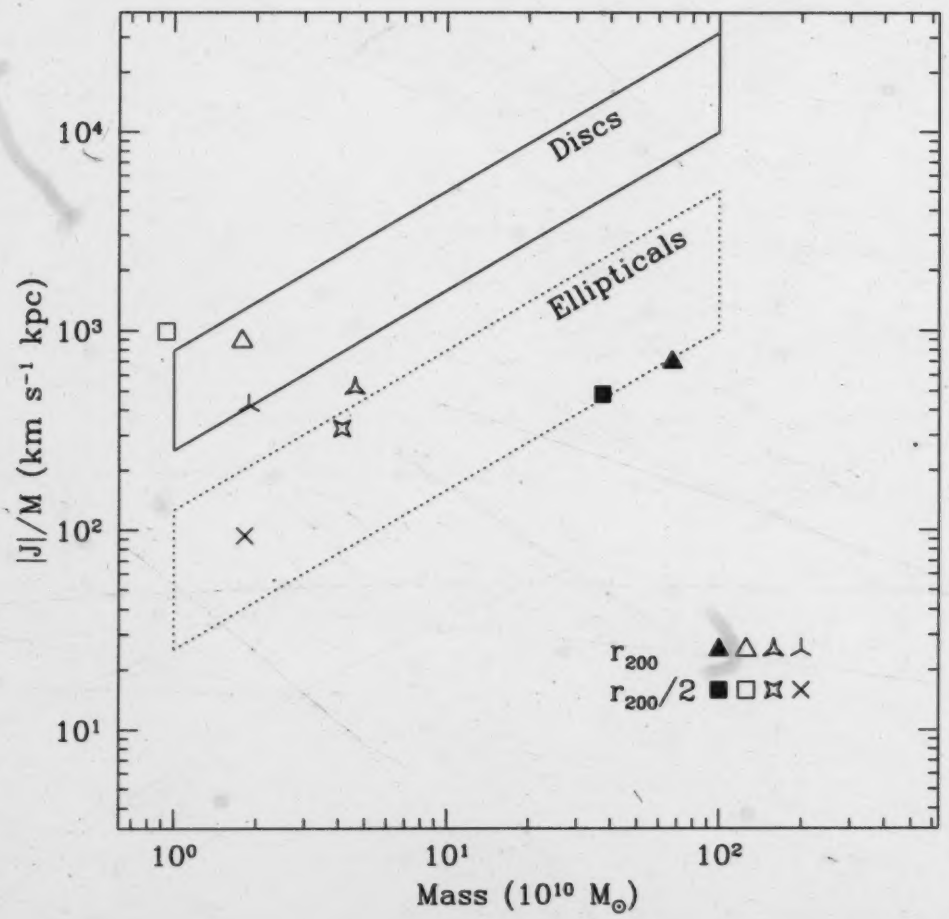

Figure 5.12: Specific angular momenta for the gas cores, hot halo, dark matter halo and star particles, within r_{200} of the main halo. Solid symbols denote the dark matter halo, open symbols the hot gas halo (all gas for which $\rho_{gas}/\rho_B < 2000$), pointed stars the gas cores (all gas for which $\rho_{gas}/\rho_B \geq 2000$) and centrally connected stars correspond to the star particles. The values for the stars and gas cores are higher than might be expected at r_{200} since a large satellite (with a significant amount of orbital angular momentum) distorts the values somewhat. The values at $r_{200}/2$ do not include this contribution, as the stellar component has a much lower specific angular momentum. Notably, the gas cores contain a higher specific angular momentum relative to the dark matter than found in the low resolution studies.

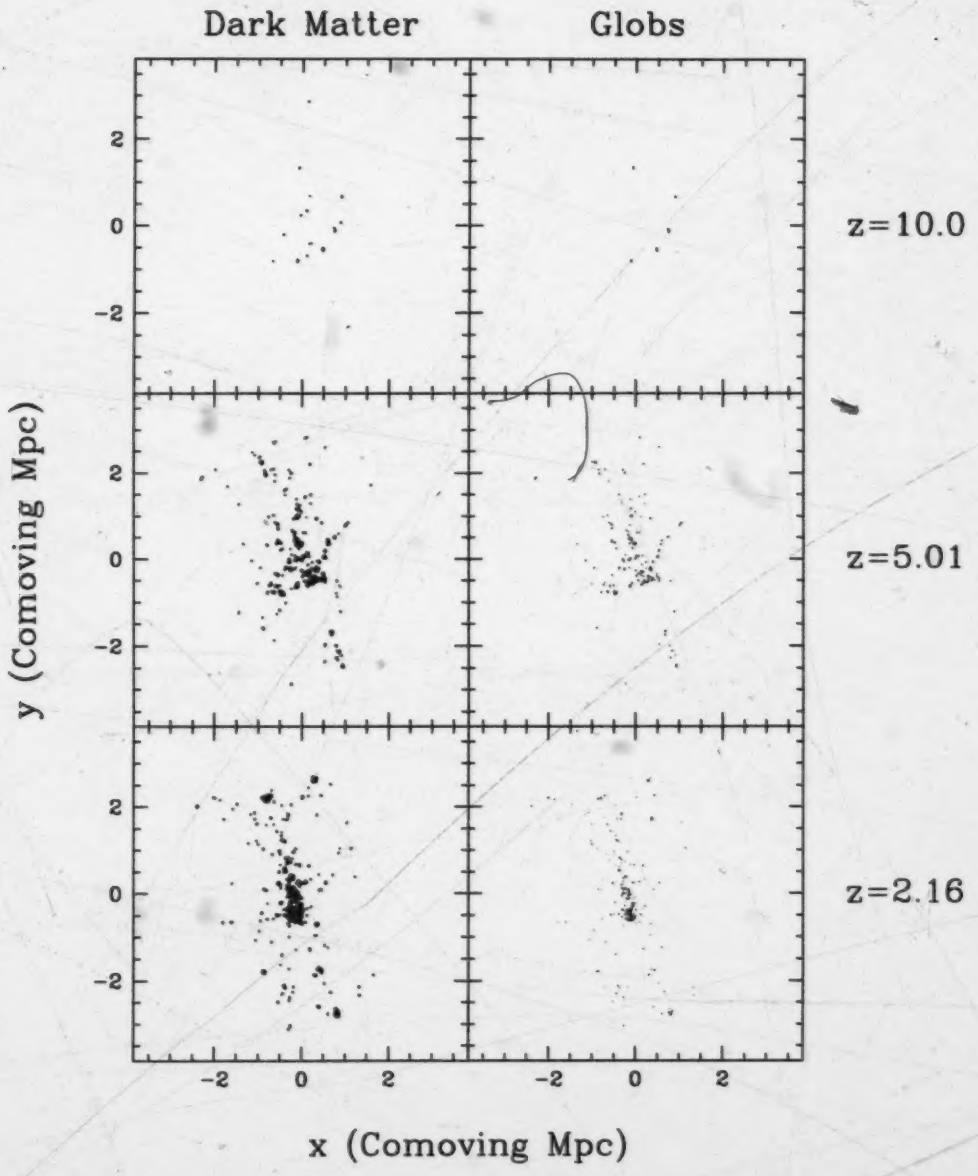

Figure 5.13: Groups found using the 'friends-of-friends' group identification algorithm at $z = 10.0, 5.01, 2.16$. Since the gas cores are more condensed than the dark matter halos in which they reside, a shorter linking length is used (see text).

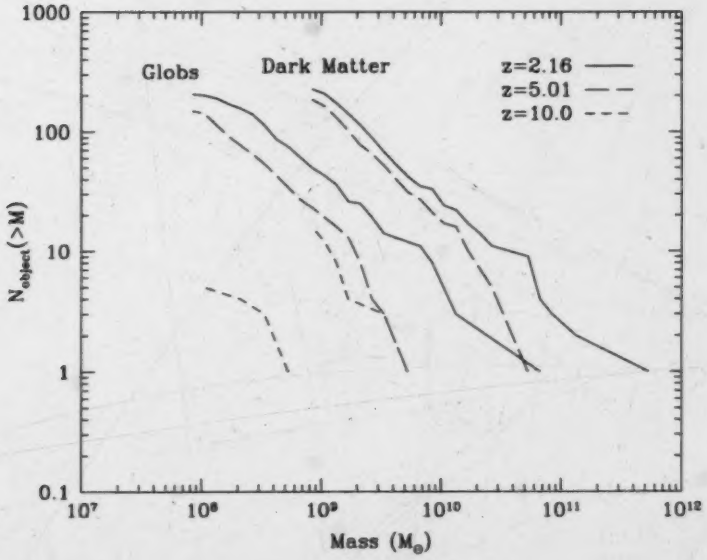

Figure 5.14: The mass multiplicity function for halos and globs in the simulation, corresponding to the groups shown in Figure 5.13. The vertical axis plots the number of halos with masses above the mass threshold given by the horizontal axis. The width of the mass bins is 0.1 dex. Comparison to a Press-Schechter prediction is not warranted since the groups are found near a high density peak, and are not representative of the average multiplicity function (which Press-Schechter theory calculates).

an integrated SFR that is less burst-like than the low resolution results in chapter 3. (Note that entirely different initial conditions were used in this simulation compared to the lower resolution one, it is not simply a higher resolution version of it.)

Althought the onset of star-formation is earlier in the high resolution run, the gradient of the SFR versus time does not rise as steeply as the low resolution runs. This is because the SFR normalization, c^*, was only 60% of that in the low resolution simulations. The peak SFR is reached at $z = 2.18$ and is 80 $M_\odot$ yr^{-1}. Had the same normalization been kept, then a linear scaling of the peak value suggests an SFR of over 100 $M_\odot$ yr^{-1} would have been found. The 80 $M_\odot$ yr^{-1} value is surprisingly similar to the maximum value for the smoothed low density data. The formation of the first star particles, and hence the first feedback events, occurred at $z = 3.4$ which is only slightly earlier than the low resolution runs ($z = 3.0$). This is simply because of the lower SFR normalization.

The low resolution SPa, SPna, and TSa models in chapter 3, all showed a sudden drop in the SFR following the first burst of star formation when compared to the no feedback run. This was attributed to feedback regions heating up the ISM and preventing star formation. The ESa and ESna runs showed no drop in the SFR and had very similar results to the run with no feedback. The high resolution run presented here, which uses the ESa algorithm, also does not show a sudden drop in the SFR following the first feedback events. Hence it appears that even with the increased mass resolution of this simulation, ESa does not have any significant effect on the SFR.

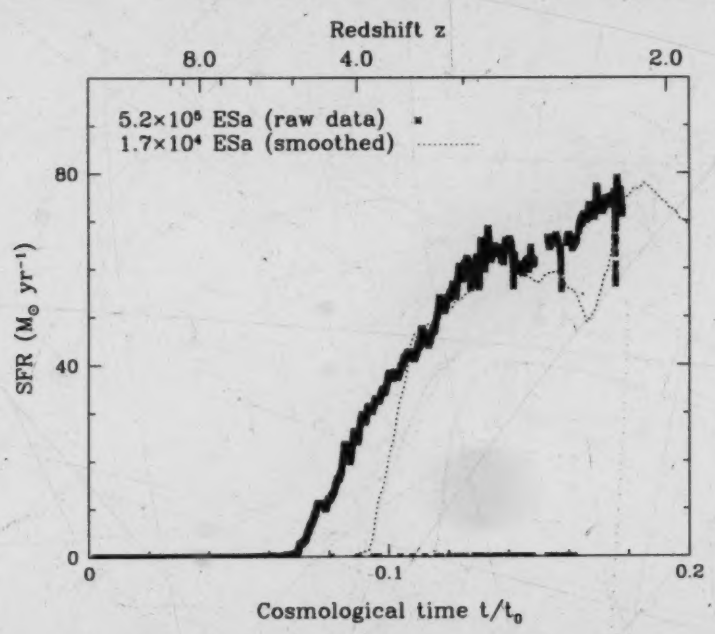

Figure 5.15: SFR integrated over the entire high resolution region. The higher mass resolution present in the simulation leads to a smoother SFR and the raw data are shown. Even with the self-gravity criterion in place, star formation begins at $z = 5$, compared to the $z = 3$ epoch for the low resolution simulations. The smoothed results from the low resolution ESa simulation in chapter 3 are shown for comparison. By $z = 2.18$ (t=0.17) the SFR has risen to 80 $M_\odot$ yr^{-1}. Dots along the bottom are related to the restarts of the code. There is a small gap in the data since some of the output files were lost due to the auto-eraser on the UK-CCC supercomputer erasing them.

5.4.5 Halo density profile

At the final stage of the simulation, $z = 2.16$, a merger is going on in the core of the halo. Thus, at least in the core region, there has been insufficient time to form a fully viralized system. Nevertheless, it is still interesting to plot the radial density profile of the system since there is much debate about

the shape of such profiles (Moore *et al.*, 1998; Primack *et al.*, 1998; Huss *et al.*, 1999; Navarro *et al.*, 1997). Recent results (Moore *et al.*, 1998; Moore *et al.*, 1999; Tissera and Dominguez-Tenreiro, 1998; Nusser and Sheth, 1999) have indicated that the slope in the core of halos is proportional to $r^{-1.4}$, which is steeper than the r^{-1} value predicted by the NFW model,

$$\rho(r) = \frac{\rho_0}{(r/r_s)[1+(r/r_s)]^2}. \tag{5.2}$$

The results of Moore *et al.* are better fitted by the following profile,

$$\rho(r) = \frac{\rho_0}{(r/r_s)^{1.4}[1+(r/r_s)^{1.4}]}, \tag{5.3}$$

which has the asymptotic property, $\rho \propto r^{-2.8}$ for $r \to \infty$, which is not the same as the $\rho \propto r^{-3}$ profile of the NFW model. The central slope is $\rho \propto r^{-1.4}$. To match the $\rho \propto r^{-3}$ profile the exponent in the $1+(r/r_s)^{1.4}$ term needs to be changed to 1.6.

Conversely, results from simulations run with the "Adaptive Refinement Tree" code of Kravtsov (1997) indicate a much *flatter* central profile, $\rho \propto r^{-0.2}$ (Primack *et al.*, 1998). Such a value is actually much closer to the observational data than that of the NFW or Moore *et al.* model (McGaugh, 1998). However, there are question marks over this result since it is not in agreement with the rest of the numerical work.

The densities of the dark matter and gas (measured using spherical radial binning) are shown in Figure 5.16. A fit of the dark matter to the Moore *et al.* profile is shown for reference. Clearly, external to a radius of 2ϵ the fit is excellent, *i.e.* the halo does not converge toward the predicted $\rho \propto r^{-1}$ line of the NFW model, and the exterior slope is shallower than $\rho \propto r^{-3}$. This may be due to the bins outside r_{200} smoothing over the central filament. Interior to 2ϵ the slope is not accurate since force softening affects the profile. The fact that it continues to match the profile so well may be a result of the baryon core 'dragging' the dark matter inward. The gas density rises above the dark matter at radii within the softening length—this was observed in chapter 3 in the low resolution simulations. Although not shown, the averaged SPH density peaks about half an order of magnitude higher than the dark matter density, equivalently half an order of magnitude lower than the bin averaged gas density. As measured by the averaged SPH density, the self-gravity criterion ($\rho_g > 0.4\rho_{dm}$), is achieved out to 4ϵ. This indicates that a large fraction of the condensing gas is available for star formation.

5.4.6 Overmerging

In Figure 5.17, the distribution of dark matter, gas and stars at $z = 2.16$ is shown. As might be expected, given that the gravitational force is only resolved to $0.02\,r_{200}$, the dark matter exhibits overmerging. This is seen by comparing the structure of the gas to that of the dark matter. The dark matter is comparatively featureless while the gas shows a number of dense cores. The softening length could have been reduced to 1 kpc, which is the 0.1 grid spacing limit, although this would probably not change the results significantly. The results of Moore *et al.* (1998) suggest that to avoid overmerging, a force resolution of $0.002\,r_{200}$ is necessary, *i.e.* ten times smaller than that used here. It is of interest to point out that Moore *et al.* have a very small softening length to initial grid spacing ratio of 1/50. Given this small ratio, it is unclear whether two-body interactions might possibly have some effect on the orbits of particles in these simulations.

5.4.7 Halo temperature profile

The temperature profile, in radial bins of size 208 particles, for the hot gas halo at $z = 2.16$ is shown in Figure 5.19. The cold gas in the progenitor halos is ignored since only gas for which $\delta_{gas} < 2000$ is included in the binning process. This procedure effectively treats the gas as being multi-phase. Variation within the bins is indicated by the $\pm 0.5\bar{\sigma}$ lines plotted above and below the profile. Since the sound crossing time is 0.03 Gyr, the gas distribution should be relaxed. To aid understanding of the temperature values, the distribution of the gas in the temperature-density plane is shown in Figure 5.18. Since the density of the halo is $n_B < 10^{-4}$ cm^{-3} and the temperature of the halo (in the central region) 6×10^5 K, the cooling times are of order 10 Gyr. Thus cooling does not yet have a significant effect on the evolution of the profile. It may not even do so *before* $z = 0$.

The central temperature is close to 10^6 K. As has already been mentioned, there are a few (order 10) particles for which $T > 10^7$ K but these are averaged over within the bin. The temperature profile is approximately flat, *i.e.* isothermal, with a temperature of 6×10^5 K out to a radius of

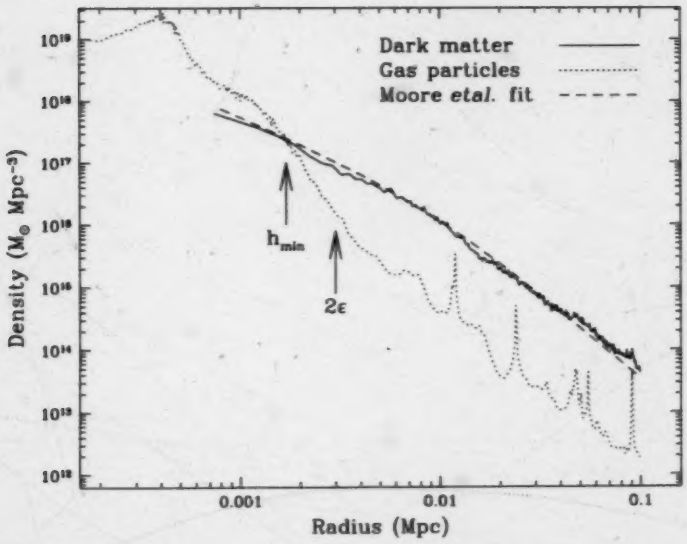

Figure 5.16: Density profiles for the dark matter and gas in the main halo. The profiles are constructed using 208 particle Lagrangian bins. There is no evidence for convergence to a $\rho \propto r^{-1}$ profile (as predicted by the NFW model). The central region is much better fitted by $\rho \propto r^{-1.4}$ as is indicated by the excellent fit provided by the Moore *et al.* (1998) profile. The outer region (beyond r_{200}) is not as well-fitted. However, the bins at this radius smooth over the large filament, which yields a higher density value than an isolated halo would have.

150 kpc. The temperature declines steeply at a radius of 250 kpc, where it falls from 2×10^5 K to $10^4 K$. The 10^4 K corresponds to the end of the cooling curve, which, in turn, corresponds to the temperature of the shocked gas accreting on to the main filament.

Isothermal models for the gas distribution in clusters (*e.g.* Cavaliere and Fusco-Femiano, 1976) are popular since the resulting density profile for gas, which is assumed to reside in an external King (1972) potential, is given by the simple analytic form,

$$\rho_{gas}(r) = \rho_0 \left[1 + \left(\frac{r}{r_c}\right)^2\right]^{-3\beta/2}, \tag{5.4}$$

where the β parameter is given by

$$\beta = \frac{\mu m_p \sigma_{dm}^2}{3kT}. \tag{5.5}$$

This value corresponds to the ratio of the specific dark matter kinetic energy (measured using the one dimensional velocity dispersion $\sigma_{1d} = \sigma_{3d}/\sqrt{3}$) to the thermal energy of the gas. Values greater than one suggest that bulk motions must support the gas in some way. Adiabatic models, which can be derived very simply from the hydrostatic equation, are not valid since most of the thermal energy is derived from non-adiabatic, *i.e.* shock, heating. A plot of the radial entropy profile shows that the edge of the high temperature region is usually associated with a radially propagating shock wave (Evrard, 1990; Thomas and Couchman, 1992). Given that plotted temperature profile is roughly isothermal within r_{200}, it was decided to calculate the β parameter for this profile. The full three dimensional velocity dispersion for the dark matter within r_{200} was found to be 163 km s^{-1}. It was also decided to calculate the β value for the entire gas population (including the cold cores). Since this lowers the overall temperature, it would be expected that β should rise. The average temperature for all of the gas within r_{200} is 2.4×10^5 K, while for the hot halo it is 6.4×10^5 K. This leads to β_{all}=1.89 and $\beta_{halo} = 0.73$. The value for the halo gas is lower than one which might

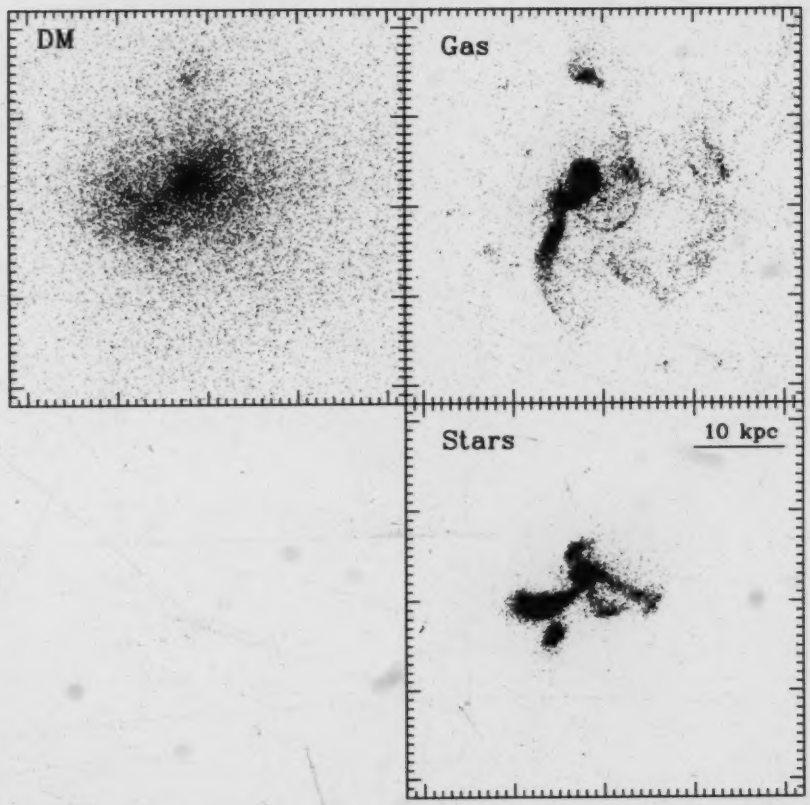

Figure 5.17: Projection of the dark matter, gas and stellar distributions in the main halo at $z = 2.16$. Overmerging is visible in the dark matter since it is not possible to associate halos directly with the stellar and gaseous features. This result is not surprising since Moore *et al.* (1998) argue that the force must be resolved to $0.002r_{200}$ to resolve this problem. This is ten times smaller than the current value.

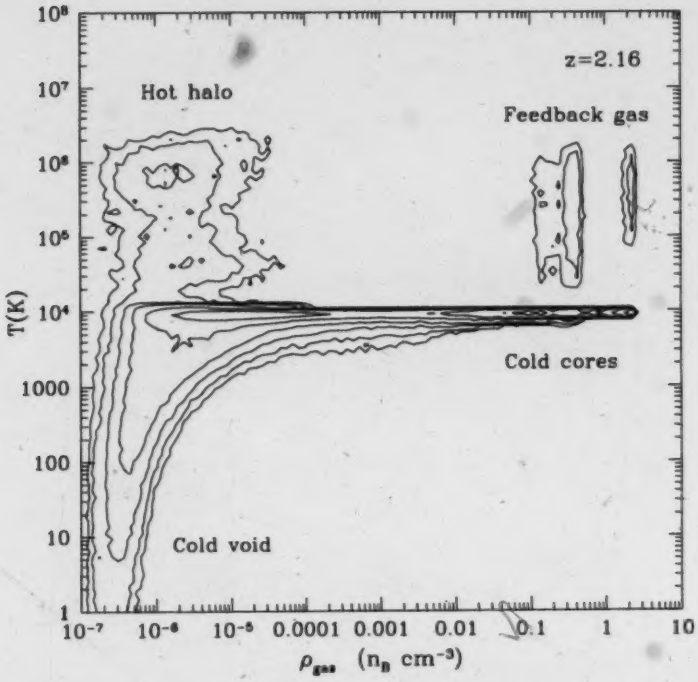

Figure 5.18: Distribution of the gas within the temperature-density plane at $z = 2.16$. The particle values are binned into a grid of size 100×100. The highest contour is set at 0.05 of the total gas mass per grid and the spacing between contours is 2.09. The lowest contour thus represents only 3×10^{-5} of the total gas mass per grid bin. The plot is very similar to previous studies, with the exception of the feedback region on the right-hand side of the plot.

be explained by energy input from feedback. To provide a value for comparison, the Santa Barbara cluster simulation (see chapter 4) has $\beta \simeq 1.2$.

5.5 Conclusion

This chapter has presented results for a simulation with sufficient mass resolution to accurately resolve shock structures within the forming galaxy. Unfortunately, due to limitations in the simulation code, it was necessary to truncate the evolution of the system at $z = 2.16$.

Principal conclusions follow.

1. Even though the simulation was stopped at $z = 2.16$, it was still possible to make useful predictions about the effect of higher resolution in simulations of galaxy formation. In particular, visualization revealed that the first baryon cores are in place by $z = 10$.

2. The cooling catastrophe continues to be a significant problem. The dwarf galaxies, even those containing greater than 10^4 gas particles, collapsed to a size close to the gravitational softening of the simulation. However, the dwarfs did not exhibit a very tight central concentration of gas which was observed in the low resolution models presented in chapter 3.

3. Even given the far higher mass resolution in this simulation than in those presented in chapter 3, ESa feedback does not appear to have a significant effect on SFR. Due to the higher resolution, the SFR for the simulation was smoother.

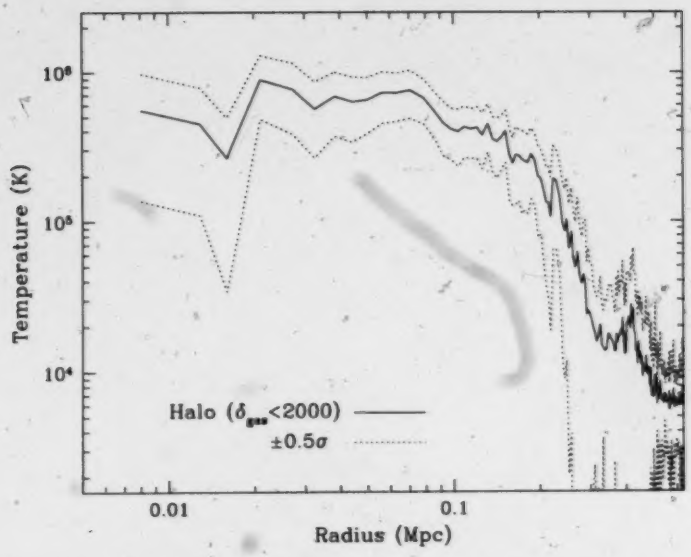

Figure 5.19: Temperature profile for the main halo at $z = 2.16$, including $\pm 0.5\sigma$ errors. The central profile is approximately isothermal, with an average temperature of 6.4×10^5 K, out to a radius of 150 kpc.

4. Overmerging is still observed. Including the baryons does not have any significant effect on this problem. In the core of the main halo, a number of the baryon cores do not have an accompanying dark matter halo.

5. Tidal tails are a very common feature. During merger events in-falling satellites often become spatially extended. One of the dwarf system exhibited a ring-like feature as a result of this process.

6. The core-halo angular momentum transport problem continues to plague this high resolution model. The star particles, which are formed from the gas cores, showed a noticeable loss of specific angular momentum relative to the dark matter.

7. To produce very early star formation, such as that suggested by the recent deep z observations (see below), either a relaxation of the self-gravity criterion is necessary or more resolution.

8. The mass multiplicity function for both the dark matter halos and globs is well fit by an M^{-1} power law. Because the function is measured near to high peak in the density field, it is tilted towards higher higher mass halos. Although it is tempting to suggest this change is due to feedback reducing the number of low mass objects, this is not the case.

9. Visualization of the simulation shows that, as was expected, most of the gas falling into the halo does so as part of a dense baryon core. For galaxy formation in CDM, smooth in-fall can only contribute a very small fraction to the final mass of a galaxy.

In retrospect, the conservative approach of reducing the SFR normalization was probably not the right one to take. It would have been better to examine what happens when a feedback was more prevalent. Even though the higher resolution moved the onset of star formation to $z = 5$, this is later than some of the observations suggest. For example, Chen *et al.* (1999) report the possible identification of a star-forming galaxy at $z = 6.68$. The estimated star formation rate is 70 $M_\odot$ yr^{-1}, assuming a flat, $h = 0.5$ cosmology. Given the simulation just presented, this result seems remarkable. Even though the self-gravity prevents the star formation until comparatively late, it is difficult to see how, with the power spectrum used, such an object could be formed. The Lyman break galaxies (see Steidel *et al.*, 1998), characterized by masses of a few 10^{10} $M_\odot$, and star

formation rates of order 5-25 $M_\odot$ yr^{-1}, are quite well fitted by the simulation presented. Both of the dwarfs, DW1 and DW2, could conceivably be likened to these objects.

Due to the marked slowdown in the algorithm, it is not feasible to integrate the simulation any further with the current algorithm. Changes, detailed in chapter 4, need to be made which may take somewhile to implement. Nevertheless, the simulation has been able to make a number of very useful predictions. Perhaps the most significant outcome is the following; even at a mass resolution of 1.5×10^6 $M_\odot$, a plausible feedback algorithm still fails to prevent the cooling catastrophe.

Chapter 6

Conclusion

"I don't pretend we have all the answers. But the questions are certainly worth thinking about."

–Arthur C. Clarke

Detailed conclusions are presented at the end of each of the preceding chapters. For completeness, the most significant conclusions from each chapter are reviewed in the context of the entire thesis. Prospects and suggestions for future work are also made.

6.1 Effect of different implementations of SPH

It was demonstrated in chapter 2 that a Monaghan-type artificial viscosity is necessary to provide reasonable accuracy in problems with small N. This result was shown to be true for a number of simulations of different physical phenomena which included shock tubes, spherical collapse and disk formation. Implementations that use a divergence-based viscosity are less accurate since they allow interpenetration of particles, and hence, exhibit poorer shock capturing. The performance of an artificial viscosity can, therefore, be viewed as being relative to its effective resolution: Monaghan-type viscosities work on a particle-particle basis while divergence-based viscosities work relative to the smoothed velocity field.

The role of different symmetrizations in the equation of motion was seen to have a less significant effect. Results tended to indicate that kernel symmetrization is the preferred method (better results for energy evolution and angular momentum conservation) but there is no strong argument against any of the methods. This result inspires confidence since SPH simulations cannot be discredited on the basis of the particular symmetrization adopted. In view of this, the symmetrization of TC92 was supplemented by kernel symmetrization in the HYDRA code, because it provided a robust and highly efficient scheme. Shear correction was seen to reduce the angular momentum transport problem at the cost of only marginally reduced shock capturing. The efficiency of the TC92 symmetrization was aided by an implementation of a Monaghan-type viscosity which did not require the precomputation of all density values. This result is particularly important since it allowed the SPH calculation to be done in one loop over the particles rather than two. The second loop was compressed into the first one by storing the list of neighbours necessary for the calculation of the energy and equation of motion. This speed improvement is particularly valuable in galaxy formation simulations which require many thousands of time-steps.

Accurate results, relative to those from modern shock-capturing methods such as PPM, require the use of a large number of particles. For spherical collapse, it was shown that at least 30,000 particles were necessary to reproduce the energy profile of the PPM simulation, and even at this particle resolution, the radial profiles showed less well-defined shocks. This result is potentially very worrying since it implies that almost all the galaxy formation studies conducted to date have poor resolution. In particular, it was demonstrated that claims of resolution of spiral features are misplaced since different SPH algorithms produced dissimilar spiral structures at the same epoch.

6.2 Modelling star formation and feedback

The inclusion of star formation and feedback is necessary to produce realistic simulations of cosmic structure formation. Although the modelling is fraught with difficulties, it should be remembered

that the aim of these investigations is to reproduce star formation in bulk (on scales associated with giant molecular clouds), and not to predict star formation on parsec scales.

Star formation was implemented in the models using a Schmidt Law to calculate the SFR for each particle. A number of different implementations of thermal feedback were considered, and it was shown that varying results can be obtained depending on the algorithm. In broad agreement with the results from other numerical work, it was found that high-resolution model galaxies are significantly affected by feedback. Both blow-out, where gas is ejected into the halo and blow-away, where it is unbound from the potential well, were observed. Furthermore, the lower mass dwarf model was intrinsically more sensitive to feedback; it exhibited larger feedback 'bubbles' and more velocity dispersion when compared to the Milky Way prototype. This is to be expected since the escape velocity is lower while the temperature of the feedback regions remains constant in the lower mass model. Disc systems formed in low-resolution cosmological simulations did not exhibit significant differences among feedback versions. Rotation curves and density profiles were broadly similar for all algorithms. However, a small, but noticeable, trend toward higher specific angular momentum was visible for the gas core with increasing feedback. This result is encouraging, but it was still not possible to form a disc system with an angular momentum value similar to that observed in disc galaxies.

Attempts to limit the radiative cooling using a revised cooling mechanism were not particularly successful. Allowing the feedback region to be adiabatic for a brief period proved to be more effective. Since in cosmological simulations a large accumulation of matter occurs within h_{min} the density of particles in feedback regions hardly changes at all. Changing the short-range neighbour search algorithm so as not to smooth over all the neighbours within h_{min} changed the SFR quite noticeably. Star formation began earlier and the SFR was roughly half that of the standard implementation. This result shows that even before changes in the star formation algorithm are considered, changes to the SPH algorithm can affect the SFR. Overall, the sensitivity of the models to changes in the parameter space is a cause for concern. Ideally, to provide comparison between algorithms, a standard test-bed simulation should be adopted. It is unclear what form this test case should take due to the different simulation algorithms currently used (for example, some are finite-difference-based while others are particle-based).

The SFRs derived from the cosmological simulations are in broad agreement with observations, but without better observations of deep z objects, it is difficult to make concrete predictions. Currently, the estimates of the star formation rates from Hβ emission differ by as much as an order of magnitude from the estimates from UV flux. Furthermore, the effects of dust extinction are not well understood. In the simulations, variation with changing feedback algorithm was noted, and the SP and TS algorithms were both capable of a strong suppression of the SFR following the first initial burst. On the basis of the isolated galaxy simulations, the ESa algorithm was preferred, however, in cosmological simulations it had little effect on the resulting disc. The self-gravity criterion was found to limit the formation of stars until quite late, $z \simeq 3$, while removing it led to unrealistically early star formation, at $z \simeq 9$. Observations currently suggest that the global star formation rate is approximately constant from $z = 1$ to $z = 4.5$ which is at odds with the results found. An accurate calculation of when star formation begins in earnest requires higher resolution simulations than those discussed.

6.3 Parallel methods

The parallelization of complex simulation codes is a difficult and arduous task even on shared memory supercomputers. Shared memory systems allow a simple programming model where only the distribution of the calculation must be addressed, as opposed to distribution of calculation and data which is required in the distributed memory model. Use of the OpenMP API allows the writing of codes which can be ported across all SMP machines, thus avoiding the need to maintain differing sets of parallel instructions.

It was demonstrated that even before parallelization, significant speed-ups could be found by revising the list structures used to address the particle data. By changing from a linked list structure to an ordered list, the execution time for the final steps of an example simulation were reduced by over 30%. The implementation of ordered particles provided even greater returns with the speed of the code more than doubling during the final evolution of the system. The overhead of sorting particles was more than outweighed by the reduced cache-miss ratio in the short range calculation. This result might be considered surprising, but in view of the penalty of a cache miss relative to the speed of floating point operations (40:1), it should not be.

Speed-ups for a simulation with 2×64^3 particles were shown to be excellent, given the difficulty in load balancing gravitational n-body calculations. For 8 PEs, the efficiency was initially 77% and it fell to 59% in the clustered state. For a 2×128^3 simulation on 16 PEs, the cycle time for a 2×128^3 simulation was 20 seconds initially, falling to 79 seconds at the end of the simulation. Since

the cycle time is only a little over a minute, simulations with many thousands of time-steps can be conducted within a time-span of one week. The cause of the inefficiency in the clustered state is the placement of adaptive refinements which requires compromises in the parallel code. Large refinements are calculated across the entire machine while smaller refinements are distributed as a task farm among the PEs. Aside from this issue, all the routines with a large work quantum were efficiently parallelized. To achieve a good load balance in some difficult situations, the dynamic scheduling option provided by the OpenMP instruction set was heavily utilized.

A few problems relating to the architecture of the Origin 2000 systems were noted. Of particular concern, is that the NUMA architecture can lead to severe contention for memory pages if references occur to the memory on a single node. This results when a large distribution of data is assigned using a block distribution, and is then reused with a smaller array size which is contained within one memory node. To partially alleviate this problem, a block cyclic distribution of arrays was used. It was also observed that using shared common blocks to reduce memory usage can lead to severe slow-down, since the memory placement for one array is often less than optimal for another.

The scaling of the code was tested up to 128 processors. This is currently the maximum commercially available size of a single image Origin 2000 system (although experimental 256 node machines do exist). It was shown that a simulation of size 2×256^3 could conceivably be carried out on this many PEs. Parallel efficiency in the unclustered state was 40%, while in the clustered state it fell to 20%. Given the high communication-to-computation ratio of the code, these results are excellent. A loss of efficiency was noted due to a necessary compromise in the way refinements are treated. Suggestions for overcoming this problem were made, although for a complex code of this type, the best parallel speed-ups will always result from writing a carefully load-balanced message passing code. However, writing such a code requires man-years of work while the SMP code took little over a month to develop in its first form. It is also much easier to adopt changes to the serial algorithm in the SMP code. A simulation of size 2×256^3 is probably the largest that can be conducted in a reasonable time-frame using the SMP code. Such simulations can be used to study the formation of galaxies in a representative volume of the Universe and thus to evaluate the bias parameter.

Multiple mass simulations were also shown to be possible with the parallel code. The main drawback for this case was found to be the slow-down that resulted from the SPH algorithm in high density regions. For 16 PEs, the efficiency fell from 56% in the unclustered state to 19% in the clustered state. The cause of the slow-down was a load imbalance due to some regions in the short-range calculation taking far longer than others. This lead to almost all the PEs waiting for work while the slowest PE finish its task. However, a dark-matter-only calculation demonstrated that in the clustered state the gravity solver ***still shows good speed-up***. A simple change to the SPH solver, so that it continues to always search over N_{smooth} neighbours, should improve the performance of the current code dramatically. Also, separating the SPH solver from the same subroutine as the gravity solver should lead to improved speed-up.

When measured in terms relevant to scientific investigation, namely the utility of the code to perform simulations of interest, the performance of the code is excellent. It allows simulations of a size in between those which can be performed on workstations and those which must be conducted on massively parallel supercomputers. While parallelizing the HYDRA algorithm is a challenging task, due to the adaptive nature of the algorithm, the effort spent in doing so is worthwhile since it is almost an order of magnitude faster than tree codes.

6.4 High resolution studies

Using the multiple mass technique, a galaxy was simulated at high resolution in a representative volume of a CDM universe. The mass resolution in the gas was 1.5×10^6 $M_\odot$, which meant the baryon component of the resimulated galaxy was represented by approximately 110,000 particles. This resolution is sufficient to overcome a number of criticisms leveled at previous studies of galaxy formation using SPH. Due to limitations in the parallel code, it was not possible to integrate the simulation beyond a redshift of $z = 2.16$, in a reasonable wall-clock time.

A conservative approach was taken in the selection of the ESa feedback algorithm. This algorithm has a realistic energy budget and does not introduce unusual features, such as sudden ejection of individual particles from star forming gas regions. It was hoped that the increased mass resolution, yielding a lower escape velocity for the first star forming systems, would allow the feedback algorithm to have more effect. This was only partially observed, and the spatial extent of the largest disc systems was still close to the hydrodynamic resolution. However, the high resolution systems did not exhibit a very dense gas nucles of size $\simeq 0.1 h_{min}$, which was observed in the low feedback runs. This suggests that feedback has had a marginally increased effect.

Overmerging was still observed in the largest dark matter halo. The presence of baryon cores did not provide more stability for the in-falling dark-matter halos. It has been shown a force resolution of $0.002\, r_{200}$ is sufficient to avoid this problem in dark-matter-only simulations. Because

hydrodynamic simulations are so much more computationally costly than dark matter, it seems unlikely that hydrodynamic simulations will meet this criterion for quite some time. The simulation discussed had a force resolution of $0.02\,r_{200}$ and reducing to the value suggested would require nearly two orders of magnitude more particles. On a related note, the core-halo angular momentum transport was observed to operate at very early times. By $z = 2.16$ the stars in the main galaxy had already lost a significant proportion of their angular momentum.

6.5 Future work

The main purpose of this thesis has been to show that simulations of galaxy formation are only just reaching the point where resolution is sufficient to make detailed predictions. 30,000 particles are necessary to accurately follow structure formation. The large number of time-steps necessary to accurately integrate the hydrodynamics, in excess of 20,000 steps to z=0, mean that realistic simulations of galaxy formation are very difficult.

6.5.1 Achieving a better understanding of numerical resolution

The presence of convergence studies in a field is a strong indicator that it is a mature one. Because simulating galaxy formation is still a comparatively young field, no systematic study of convergence has been undertaken. Furthermore, until recently, there had been no convergence studies for cosmological hydrodynamic simulations at all. In view of the cooling catastrophe, which in numerical simulations means that more gas cools with increasing resolution, it is vital that a systematic study of the effect of resolution be conducted. Determining what mass resolution is required to avoid the cooling catastrophe, when feedback is included, should be the fundamental goal for the simulation field. The core-halo angular momentum transport mechanism also makes convergence studies important. Increasing resolution leads to more (smaller) halos being resolved which, in turn, suffer from angular momentum loss higher up the merger tree. Since most of the angular momentum in a hierarchical system is bound in the orbital angular momentum of merging halos it is difficult to predict the result of adding more resolution. The efficient disc formation noted in a number of early studies is a consequence of a lack of resolution.

Convergence studies provide far more understanding than just adding higher resolution and more physics. Indeed, an argument can be made that adding more physics, without understanding the underlying effects of resolution, leads to investigations which actually contribute very little. For this reason, the work presented in this thesis has been quite conservative in its incorporation of different physics. There is a noticeable trend in the simulation field to obtain as many predictions as possible by including ever more physical effects without paying careful attention to the numerics. This should be strongly discouraged and the basis of numerical methods, namely convergence, should be revisited.

If the amount of gas which cools at a given resolution can be quantified, then the development of *resolution-independent* models can begin. This will also help to better understand the calculation of the SFR since it is entirely dependent on the amount of cold gas within a simulation. The fact that a *lack of resolution* acts in a similar fashion to a feedback process has already been hinted at by a number of authors, but, as yet, it is not properly understood. Convergence tests will quantify the effect. All feedback models presented so far, including the multi-phase ones, exhibit dependence on the resolution of the simulation. This thesis has shown that the effect of feedback is strongly dependent upon the fundamental mass scale being modelled. Thus, once studies of the cooling catastrophe have been undertaken, the next step is to perform a similar series of calculations with feedback.

The question of whether a Jeans' Mass/Length need or need not be resolved in SPH simulations is as yet unanswered. In simulations of star formation the initial conditions are very smooth and only a small number of long wavelength modes are imposed as a perturbation. Star formation then occurs as the large growing modes fragment, forming binary systems for example. In contrast, simulations of hierarchical clustering begin from initial conditions that contain growing modes on all scales within the simulation. The smooth nature of the collapse in star formation simulations, means that the evolution is susceptible to small truncation errors, which occur when the Jeans' length is not resolved. These errors then grow, driven by the physics being simulated. It would be expected that in hierarchical simulations such errors, that might be present in the largest growing mode, are swamped by the physical growing modes at shorter wave lengths. This phenomenon requires detailed and thorough investigation.

6.5.2 Improvements to the simulation algorithm

There is a significant drawback in the SPH algorithm implemented in HYDRA . For high resolution systems in which baryon condensation occurs, the h_{min} limitation to the neighbour search means that it becomes possible for particles to smooth over tens of thousands of neighbour particles. This resolution is largely wasted since the objects in which this occurs are largely isothermal and comparatively featureless (at least in galaxy formation simulations). It would be useful to investigate whether an implementation that continues to search over N_{smooth} neighbours can be tuned to reproduce the results of one which smooths over all the particles within h_{min}. Since the resolution is effectively being degraded, this should be possible. Modifying the density estimate should be straightforward. For the energy equation, it is not clear how to proceed due to the summation having terms in **r.v**. Numerical calibration may be necessary. If such a modification can be made, the performance of the parallel algorithm will be greatly improved by avoiding a possibly severe load imbalance.

The current implementation of HYDRA lacks the facility for multiple time-steps. Much has been made in the literature of the utility of multiple time-steps, although this is partially a function of the use of extremely small softening lengths. For the simulations conducted with HYDRA , softening lengths typically equal to, or greater, than 1/10th of the original grid spacing are chosen. In such cases, a large enough number of particles sit close enough to the smallest time-step bin to render multiple time-steps less effective. Nonetheless, a good implementation is estimated to improve the speed of the code by a factor of three.

A number of drawbacks in the shared memory code have been indicated. The best way to circumvent these problems is to write an MPI version of the code which allows an efficient domain decomposition of the computer, especially for calculating the refinements. This work is currently underway at the Edinburgh Parallel Computing Center. However, the effort involved in writing such a code is extreme (especially considering that a load balanced version of the P^3M algorithm took a man-year to write). It is interesting to note that the distributed memory code is more conceptually elegant than the shared memory version, since the domain decomposition of the machine can be performed hierarchically. This is much more in the spirit of the algorithm than the hard division between refinements performed across the whole machine and the task farm, as in the shared-memory code. A secondary spur behind this research is that there are a number of massively parallel computer systems available to academic researchers which are comparatively under-utilized because of the difficulty in programming for them.

6.5.3 Incorporating more realistic physics

A coherent theory of star formation remains elusive. Ideally, a dialogue between star formation theorists and cosmologists performing galaxy formation simulations should be begun. The use of a simple Schmidt Law to calculate the SFR, whilst being experimentally motivated, actually exhibits an enormous scatter when compared to real data. This must be improved, and in particular improvements are not likely to come from more detailed examinations of the SPH quantities.

Chemistry and variable metallicity are beginning to appear in simulations of galaxy formation. The cooling function is very sensitive to metallicity, and at low temperatures (< 1000 K), H_2 molecular cooling becomes important. While there is little doubt that chemistry is particularly important in the formation of 'the first objects', its relevance to galaxy formation is less clear (*i.e.* is the geometry of the collapse more important?). Since there is a strong feedback loop associated with star formation, metallicity enrichment and gas cooling time, it is questionable whether this process can be modelled accurately at the low resolution of current simulations. However, both chemistry and variable metallicity have been implemented in the HYDRA algorithm and simulations that explore the effects of these phenomena will certainly need to be conducted to gain a better understanding of galaxy formation.

At present, magnetic fields are largely ignored in simulations of galaxy formation. Including them adds yet another unknown, namely the distribution of the pregalactic field. However, it is widely known that magnetic fields are vital in star formation, and hence, it is perhaps only a matter of time before they are included in galaxy formation simulations. SPH has been shown to be able to integrate magnetic fields to a reasonable accuracy ($\nabla.\mathbf{B}/|\mathbf{B}| < 0.01$) and the numerical implementation is comparatively straight forward (Dolag *et al.*, 1999).

6.5.4 Prospects for galaxy formation in the CDM model of structure formation

The standard Einstein-de Sitter CDM model is widely discredited on a number of grounds (*e.g.* bulk flows and cluster abundance when COBE normalized). With regards to galaxy formation, the cooling catastrophe presents a significant problem. Without the presence of feedback, star formation occurs catastrophically early and it is almost impossible to form disc galaxies, since too little gas is

left at the disc formation epoch ($z \simeq 1$). Furthermore, the in-fall of matter is highly inhomogeneous, contrary to the traditionally perceived idea of smooth in-fall necessary to form discs (*i.e.* along the lines of the classic Eggen *et al.*, 1962, paper). Some dynamicists, in particular Sellwood, argue that disc formation from CDM-like initial conditions is almost impossible.

Numerical simulations have shown that the angular momentum transport problem is a serious drawback for the CDM model. The core-halo problem is particularly pervasive in the CDM cosmology because of the hierarchical formation of objects. At each 'level' of the formation process, the gas cores will lose yet more angular momentum. It is difficult to draw a firm conclusion as to whether feedback is capable of solving this problem. Semi-analytic approaches make a number of assumptions that may or may not be accurate. This thesis has been able to show, at low resolution, that strong feedback does lead to a smaller angular momentum loss. However, the reduced loss was not sufficient to increase the angular momentum to that of a disk galaxy. Models, where the gas is allowed to remain adiabatic until $z = 1$ (Weil *et al.*, 1998), reproduce discs with a similar specific angular momentum to the dark matter halo. These models can be viewed as having feedback which perfectly balances radiative losses.

McGaugh (1998) presents a review of evidence that the rotation curves of CDM-type halos do not fit the observational data for low surface brightness galaxies. The halos are too centrally concentrated which leads to a sharp rise in the rotation curve which is not observed. It should be emphasized that this problem occurs even before the core-halo transport problem is considered. The idea that feedback should be capable of rearranging the dark matter structure (Navarro *et al.*, 1996*b*) does not appear to be supported by observations, and also appears to be difficult to argue on theoretical grounds (Mac Low and Ferrara, 1999). Changing to a critical-density CDM model with $1 = \Omega = \Omega_{matter} + \Omega_{\Lambda}$ produces a better fit to the data for $\Omega_{matter} = 0.3$. This is in broad agreement with observational results, in particular the Supernovae Cosmology project, suggesting that the matter content of the universe is less than 0.3 of the critical density.

6.5.5 Prospects for the simulation field

Setting aside issues relating to convergence studies, it is interesting to ponder the near future for the simulation field. The overmerging studies conducted by Moore *et al.* (1998) suggest that it is necessary to have force resolution to $0.002\, r_{200}$ to avoid the overmerging problem. This kind of force resolution requires a particle number at least an order of magnitude higher than the high resolution simulations presented in chapter 5. It will be necessary to use the MPI code under development, since these simulations will require a truly enormous number of time-steps, probably well over 100,000. In which case the wall-clock time per step cannot go beyond 10 seconds, as otherwise, the simulation will be intractable. Given the excellent scaling observed in non-adaptive SHMEM code (Macfarland *et al.*, 1998), these simulation sizes are feasible.

What would a simulation of this kind tell us? Aside from issues of convergence, the mass resolution would be 1×10^5 $M_{\odot}$, so that the very largest molecular clouds (GMCs) will be resolved. This is an exciting prospect, since the star formation will then become closely tied to the processes observed. It should also be possible to determine the effect of the local dark matter dynamics on the SFR. Of course such high resolution probably necessitates a reworking of the star formation algorithm, given that the GMCs are on the verge of being resolved. The feedback models considered in this thesis should have an enormous impact on the first structures to form in such a simulation. The results from the first generation of objects formed will also be relevant to observations from the *Next Generation Space Telescope*. Also, merger rates for the progenitor halos with be calculable and a detailed examination of the satellite population will be possible. Related to this, the effect of 'galaxy harassment' on the dwarf galaxies which merge to form the final system will be particularly noticeable. A number of parameters relevant to the semi-analytic models, such as the effect of dynamical friction, will also be constrained. Given a detailed model of additional physics, such as metal enrichment in the ISM, the effect of self-regulation on the SFR should be elucidated. Also the spectral colours of the galaxy will be predicted with great detail. At the dynamical level, if a disc forms, there should be sufficient resolution to form spiral features of the form predicted by the Lin-Shu mechanism. The non-linear *vs.* linear evolution of the angular momentum will be predicted in great detail, which will also allow the affect of the core-halo transport problem to be observed in the wider context. Finally, the simulation would, once and for all, demonstrate whether it is possible to form disc galaxies in CDM cosmologies.

Appendix A

A note regarding the artificial viscosity and energy equations in comoving coordinates

To see how the corrections applied in **accel** come about, it useful to perform the derivation of the energy equation in comoving coordinates. This also highlights a minor issue in relation to how the artificial viscosity is treated. The derivation uses x to denote comoving quantities and r for physical coordinates. Where unclear, a c subscript is used to denote comoving quantities. u denotes internal energy, W the kernel function and the remaining terms are standard.

Simple results worth noting:

$$\nabla_{\mathbf{r}} = \frac{1}{a}\nabla_{\mathbf{x}}$$

$$v = \frac{D\mathbf{r}}{dt} = a\dot{\mathbf{x}} + \dot{a}\mathbf{x}$$

$$W(\mathbf{r}_{ij}, h_i) = \frac{W(\mathbf{x}_{ij}, h_i^c)}{a^3}$$

Beginning from energy equation in physical coordinates,

$$\frac{Du}{dt} = -\frac{P}{\rho}\nabla_{\mathbf{r}}\mathbf{v}, \tag{A.1}$$

$u_{phys} = a^2 u_c$ may be used to substitute for u_{phys}, and expanding the LHS,

$$\frac{Da^2u_c}{dt} = 2a\dot{a}u_c + a^2\frac{Du_c}{dt}.$$

The SPH version of the RHS is given by

$$\frac{Du}{dt} = -\frac{2}{3}\sum m_j \frac{P_i}{\rho_i^2}\Delta\mathbf{v}_{ij}.\nabla_i W(\mathbf{r}_{ij}, h_i). \tag{A.2}$$

Substituting the artificial viscosity gives

$$\frac{Du}{dt} = -\frac{2}{3}\sum m_j \left(\frac{P_i}{\rho_i^2} + \Pi_{ij}\right)\Delta\mathbf{v}_{ij}.\nabla_i W(\mathbf{r}_{ij}, h_i). \tag{A.3}$$

First substitute for $\mathbf{v}$ in comoving coordinates, then,

$$\frac{Du}{dt} = -\frac{2}{3}\sum m_j \left(\frac{P_i}{\rho_i^2} + \Pi_{ij}\right)(\dot{a}\,\Delta\mathbf{x}_{ij} + a\,\Delta\mathbf{v}_{\mathbf{x}ij}).\nabla_i W(\mathbf{r}_{ij}, h_i).$$

Next substitute for ∇_i, pressure and density, to get

$$\frac{Du}{dt} = -\frac{2}{3}\sum m_j \left(a^5 \frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij} \right) (\dot{a}\,\Delta\mathbf{x}_{ij} + a\,\Delta\mathbf{v}_{\mathbf{x}ij}).\frac{1}{a}\nabla_{\mathbf{x}}\frac{1}{a^3}W(\mathbf{x}_{ij}, h_i^c).$$

The i subscript on the gradient has been dropped but is still implied. Clearly the equation splits up into a contribution from the Hubble flow and a contribution from peculiar motion:

$$\frac{Du}{dt} = -\frac{2}{3}\sum m_j \left(a^5 \frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij} \right) H\,\Delta\mathbf{x}_{ij}.\frac{1}{a^3}\nabla_{\mathbf{x}}W(\mathbf{x}_{ij}, h_i^c)$$

$$-\frac{2}{3}\sum m_j \left(a^5 \frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij} \right) \Delta\mathbf{v}_{\mathbf{x}ij}.\frac{1}{a^3}\nabla_{\mathbf{x}}W(\mathbf{x}_{ij}, h_i^c). \tag{A.4}$$

The expansion of Π_{ij} is more detailed. First calculate μ_{ij},

$$\mu_{ij} = \frac{h_{ij}\,\Delta\mathbf{v}_{ij}.\Delta\mathbf{r}_{ij}}{\Delta\mathbf{r}_{ij}{}^2 + \nu^2}. \tag{A.5}$$

Then substitute for comoving coordinates,

$$\mu_{ij} = \frac{a h_{ij}^c (\Delta\mathbf{v}_{\mathbf{x}ij}.\Delta\mathbf{x}_{ij} + H\,\Delta\mathbf{x}_{ij}{}^2)}{\Delta\mathbf{x}_{ij}{}^2 + \nu^{c^2}}.$$

For simplicity the following notation is adopted:

$$\Delta\mathbf{v}_{\mathbf{x}ij}.\Delta\mathbf{x}_{ij} + H\,\Delta\mathbf{x}_{ij}{}^2 = \Gamma_{ij},$$

$$\Delta\mathbf{x}_{ij}{}^2 + \nu^{c^2} = \theta_{ij}.$$

Π_{ij} can now be calculated in full,

$$\Pi_{ij} = \frac{-\alpha a h_{ij}^c \Gamma_{ij} c_{ij}}{\theta_{ij}\rho_{ij}} + \frac{\beta a^2 h_{ij}^{c^2} \Gamma_{ij}^2}{\theta_{ij}^2 \rho_{ij}}. \tag{A.6}$$

After substituting for the sound speed and density in comoving coordinates,

$$\Pi_{ij} = a^5 \left[\frac{-\alpha h_{ij}^c \Gamma_{ij} c_{ij}^c}{\theta_{ij}\rho_{ij}^c} + \frac{\beta h_{ij}^{c^2} \Gamma_{ij}^2}{\theta_{ij}^2 \rho_{ij}^c} \right] = a^5 \Pi_{ij}^c, \tag{A.7}$$

so clearly there is scaling relation. Next substitute back into equation A.4:

$$\frac{Du}{dt} = -\frac{2}{3}a^2 \sum m_j \left(\frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij}^c \right) H\,\Delta\mathbf{x}_{ij}.\nabla_{\mathbf{x}}W(\mathbf{x}_{ij}, h_i^c)$$

$$-\frac{2}{3}a^2 \sum m_j \left(\frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij}^c \right) \Delta\mathbf{v}_{\mathbf{x}ij}.\nabla_{\mathbf{x}}W(\mathbf{x}_{ij}, h_i^c). \tag{A.8}$$

After performing a little rearrangement, the comoving energy equation is derived:

$$\frac{Du_c}{dt} = -\frac{2}{3}H \sum m_j \left(\frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij}^c \right) \Delta\mathbf{x}_{ij}.\nabla_{\mathbf{x}}W(\mathbf{x}_{ij}, h_i^c)$$

$$-\frac{2}{3}\sum m_j \left(\frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij}^c \right) \Delta\mathbf{v}_{\mathbf{x}ij}.\nabla_{\mathbf{x}}W(\mathbf{x}_{ij}, h_i^c)$$

$$-2Hu_c. \tag{A.9}$$

The first term of equation A.9 is actually an SPH estimate of $-(2/3)Hu_c\nabla_{\mathbf{x}}.\mathbf{x}$. Thus if the identity $\nabla_{\mathbf{x}}.\mathbf{x} = 3$ is substituted in its place then an 'approximate' comoving energy equation is derived,

$$\frac{Du_c}{dt} = -\frac{2}{3}\sum m_j \left(\frac{P_i^c}{\rho_i^{c^2}} + \Pi_{ij}^c\right) \Delta\mathbf{v}_{\mathbf{x}ij}.\nabla_{\mathbf{x}}W(\mathbf{x}_{ij}, h_i^c) - 4Hu_c \qquad \text{(A.10)}$$

Tests were conducted on both of these equations of motion. Surprisingly it was found that equation A.10 leads to better energy conservation than equation A.9. Closer inspection showed that the reason for this is the removal of the scatter inherent in the SPH estimate of $\nabla_{\mathbf{x}}.\mathbf{x}$, which leads to a poor calculation of the PdV cooling due to the Hubble expansion. Since the difference between the two equations is proportional to H, the equations agree better at the end of the simulation than at the beginning. When converted back to physical coordinates, the comoving equation A.10 has an additional term proportional to the difference between $\nabla.\mathbf{r}$ and the SPH estimate of it. This term is small and has considerably less effect on the energy than the Hubble expansion.

Bibliography

Aarseth, S. J. (1963). Dynamical evolution of clusters of galaxies-I. *Mon. Not. R. astr. Soc.* **126**, 223–255.

Aarseth, S. J. (1966). Dynamical evolution of clusters of galaxies-II. *Mon. Not. R. astr. Soc.* **132**, 35–65.

Aarseth, S. J. (1969). Dynamical evolution of clusters of galaxies-III. *Mon. Not. R. astr. Soc.* **144**, 537–548.

Aarseth, S. J. (1971). Binary evolution in stellar systems. *Astrophys. & Space Sci.* **13**, 324–334.

Aarseth, S. J., E. L. Turner and J. R. Gott III (1979). N-body simulations of galaxy clustering. I - Initial conditions and galaxy collapse times. *Astrophys. J.* **228**, 664–683.

Abel, T., A. Stebbins, P. Anninos and M. L. Norman (1998). First structure formation. II - Cosmic string plus hot dark matter models. *Astrophys. J.* **508**, 530–534.

Amza, C., A. L. Cox, S. Dwarkadas, P. Keleher, H Lu, R. Rajamony, W. Yu and W. Zwaenepoel (1997). Treadmarks: Shared memory computing on networks of workstations. *preprint: Department of Computer Science, Rice University.*

Anderson, D. V. and D. E. Shumaker (1995). Hybrid ordered particle simulation (HOPS) code for plasma modeling on vector-serial, vector-parallel, and massively parallel computers. *Comp. Phys. Comm.* **87**, 16+.

Balsara, D. (1995). Von Neumann stability analysis of smoothed particle hydrodynamics – suggestions for optimal algorithms. *J. Comp. Phys.* **121**, 357+.

Barnes, J. and P. Hut (1986). A hierarchical O(N log N) force-calculation algorithm. *Nature* **324**, 446+.

Barnes, J. E. (1992). Transformations of galaxies. I - Mergers of equal-mass stellar disks. *Astrophys. J.* **393**, 484–507.

Bate, M. R. and A. Burkert (1997). Resolution requirements for smoothed particle hydrodynamic calculations with self-gravity. *Mon. Not. R. astr. Soc.* **288**, 1060–1072.

Benz, W. (1990). Smoothed particle hydrodynamics - A review. In: *Numerical Modelling of Nonlinear Stellar Pulsations Problems and Prospects* (J. Robert, Ed.). Kluwer Academic Publishers. Dordrecht. pp. 269+.

Bertschinger, E. (1992). Large-scale structures and motions: Linear theory and statistics. In: *New insights into the universe : proceedings of a summer school, 23-27 September 1991* (M. Portilla V. J. Martinez and D. Saez, Eds.). Springer-Verlag. New York.

Binney, J. and S. Tremaine (1987). *Galactic Dynamics.* Princeton University Press. Princeton, NJ.

Blanton, M., R. Cen, J. P. Ostriker, M. A. Strauss and M. Tegmark (1999). Time evolution of galaxy formation and bias in cosmological simulations. *astro-ph/9903165.*

Blumenthal, G. R., S. M. Faber, J. R. Primack and M. J. Rees (1984). Formation of galaxies and large-scale structure with cold dark matter. *Nature* **311**, 517–525.

Bond, J. R. and A. S. Szalay (1983). The collisionless damping of density fluctuations in an expanding universe. *Astrophys. J.* **274**, 443–468.

Bond, J. R. and G. Efstathiou (1984). Cosmic background radiation anisotropies in universes dominated by nonbaryonic dark matter. *Astrophys. J. Lett.* **285**, L45–L48.

Bond, J. R., L. Kofman and D Pogosyan (1996). How filaments are woven into the cosmic web. *Nature* **380**, 606+.

Bond, J. R., S. Cole, G. Efstathiou and N. Kaiser (1991). Excursion set mass functions for hierarchical gaussian fluctuations. *Astrophys. J.* **379**, 440–460.

Bonnor, W. B. (1956). *Z. Astrophys.* **39**, 143+.

Book, D. L. and J. P. Boris (1973). Flux-corrected transport I. SHASTA, a fluid transport algorithm that works. *J. Comp. Phys.* **11**, 38–69.

Bottema, R. and J. P. E. Gerritsen (1997). An investigation of the structure and kinematics of the spiral galaxy NGC 6503. *Mon. Not. R. astr. Soc.* **290**, 585–598.

Brieu, P. P. and A. E. Evrard (1998). P4M: A parallel version of P3M. *American Astronomical Society Meeting* **192**, 4202+.

Bruzual, A. G. and S. Charlot (1993). Spectral evolution of stellar populations using isochrone synthesis. *Astrophys. J.* **405**, 538–553.

Bryan, G. L. and J. V. Kepner (1998). Simulating galaxy formation. In: *Abstracts of the 19th Texas Symposium on Relativistic Astrophysics and Cosmology, held in Paris, France, Dec. 14-18, 1998* (J. Paul, T. Montmerle and E. Aubourg, Eds.). CEA Saclay. Paris. pp. E586+.

Burger, P. (1965). Theory of large-amplitude oscillation in the one-dimensional low-pressure cesium thermionic converter. *J. Appl. Phys.* **36**, 1938–1943.

Cavaliere, A. and R. Fusco-Femiano (1976). X-rays from hot plasma in clusters of galaxies. *Astron. Astrophys.* **49**, 137–144.

Cen, R. and J. P. Ostriker (1998). Physical bias of galaxies from large-scale hydrodynamic simulations. *astro-ph/9809370.*

Chen, H. W., K. M. Lanzetta and S. Pascarelle (1999). Spectroscopic identification of a galaxy at a probable redshift of z = 6.68.. *Nature* **398**, 586–588.

Christodoulou, D. M., I. Shlosman and J. E. Tohline (1995). A new criterion for bar-forming instability in rapidly rotating gaseous and stellar systems. 1: Axisymmetric form. *Astrophys. J.* **443**, 551–562.

Cole, S., A. Aragon-Salamanca, C. S. Frenk, J. F. Navarro and S. E. Zepf (1994). A recipe for galaxy formation. *Mon. Not. R. astr. Soc.* **271**, 781+.

Coles, P. and F. Lucchin (1995). *Cosmology. The origin and evolution of cosmic structure.* Wiley. Chichester.

Colin, P., A. Klypin, Kravtsov A. and A. Khokhlov (1998). Evolution of bias in different cosmological models. *astro-ph/9809202.*

Collela, P. and P. R. Woodward (1984). The piecewise parabolic method (PPM) for gas-dynamical simulations. *J. Comp. Phys.* **54**, 174–201.

Connolly, A. J., A. S. Szalay, M. Dickinson, M. U. Subbarao and R. J. Brunner (1997). The evolution of the global star formation history as measured from the Hubble Deep Field. *Astrophys. J. Lett.* **486**, L11+.

Contardo, G., M. Steinmetz and U. Fritze-Von Alvensleben (1998). Photometric evolution of galaxies in cosmological scenarios. *Astrophys. J.* **507**, 497–506.

Copi, C. J., D. N. Schramm and M. S. Turner (1995). Big bang nucleosynthesis and a new approach to galactic chemical evolution. *Astrophys. J. Lett.* **455**, L95+.

Couchman, H. M. P. (1991). Mesh-refined P3M - a fast adaptive N-body algorithm. *Astrophys. J. Lett.* **368**, L23–L26.

Couchman, H. M. P. (1992). Cosmological simulations using particle-mesh methods. *preprint: Department of Astronomy, University of Western Ontario.*

Couchman, H. M. P., F. R. Pearce and P. A. Thomas (1996). Hydra code release. *astro-ph/9603116.*

Cox, A. L., Y. C. Hu, H Lu and W. Zwaenepoel (1999). OpenMP on networks of SMPs. In: *Proceedings of the Thirteenth International Parallel Processing Symposium, April 1999.*

Dave, R., J. Dubinski and L. Hernquist (1997). Parallel TREESPH. *New Astronomy* **2**, 277–297.

Davis, M., G. Efstathiou, C. S. Frenk and S. D. M. White (1985). The evolution of large-scale structure in a universe dominated by cold dark matter. *Astrophys. J.* **292**, 371–394.

de Sitter, W. (1917). *Mon. Not. R. astr. Soc.* **78**, 3+.

Decyk, V. K., S. R. Karmesin, A. de Boer and P. C. Liewer (1996). Optimization of particle-in-cell codes on risc processors. *Comp. Phys. Comm.* **10**, 290+.

Dolag, K., M. Bartelmann and H. Lesch (1999). SPH simulations of magnetic fields in galaxy clusters. *astro-ph/9906329.*

Dominguez-Tenreiro, R., P. B. Tissera and A. Saiz (1998). Disk formation in hierarchical hydrodynamical simulations: A way out of the angular momentum catastrophe. *Astrophys. J. Lett.* **508**, L123–L127.

Eddington, A. S. (1924). *The Mathematical Theory of Relativity.* Cambridge University Press.

Efstathiou, G. and J. W. Eastwood (1981). On the clustering of particles in an expanding universe. *Mon. Not. R. astr. Soc.* **194**, 503–525.

Eggen, O. J., D. Lynden-Bell and A. R. Sandage (1962). Evidence from the motions of old stars that the galaxy collapsed. *Astrophys. J.* **136**, 748+.

Einstein, A. (1917). Cosmological considerations on the general theory of relativity (trans). *S.-B. Preuss. Akad. Wiss. Sitzber.* pp. 142–152.

Eke, V. R., S. Cole and C. S. Frenk (1996). Cluster evolution as a diagnostic for Omega. *Mon. Not. R. astr. Soc.* **282**, 263–280.

Elizondo, D., G. Yepes, R. Kates and A. Klypin (1999*a*). Hydrodynamical simulations of galaxy properties: Environmental effects. *New Astronomy* **4**, 101–132.

Elizondo, D., G. Yepes, R. Kates, V. Mueller and A. Klypin (1999*b*). Self-regulating galaxy formation as an explanation for the Tully-Fisher relation. *Astrophys. J.* **515**, 525–541.

Evrard, A. E. (1988). Beyond N-body - 3d cosmological gas dynamics. *Mon. Not. R. astr. Soc.* **235**, 911–934.

Evrard, A. E. (1990). Formation and evolution of X-ray clusters - a hydrodynamic simulation of the intracluster medium. *Astrophys. J.* **363**, 349–366.

Evrard, A. E. (1999). How many clusters in the sky? Expectations from the Hubble volume light-cone surveys. *American Astronomical Society Meeting* **194**, 5810+.

Evrard, A. E., F. J. Summers and M. Davis (1994). Two-fluid simulations of galaxy formation. *Astrophys. J.* **422**, 11–36.

Fall, S. M. and G. Efstathiou (1980). Formation and rotation of disc galaxies with halos. *Mon. Not. R. astr. Soc.* **193**, 189–206.

Ferrell, R. and E. Bertschinger (1994). Particle-mesh methods on the connection machine. *Int. J. Mod. Phys. C* **5**, 933+.

Ferrell, R. and E. Bertschinger (1995). A parallel processing algorithm for computing short-range particle forces with inhomogeneous particle distributions. In: *Proceedings of the 1995 Society for Computer Simulation Multiconference, astro-ph/9310002.*

Frenk, C. S., A. E. Evrard, S. D. M. White and F. J. Summers (1996). Galaxy dynamics in clusters. *Astrophys. J.* **472**, 460+.

Frenk, C. S., S. D. M. White, P. Bode, J. R. Bond, G. L. Bryan, R. Cen, H. M. P. Couchman, A. E. Evrard, N. Gnedin, A. Jenkins, A. M. Khoklov, A. Klypin, J. F. Navarro, M. L. Norman, J. P. Ostriker, J. M. Owen, F. R. Pearce, U.-L. Pen, M. Steinmetz, P. A. Thomas, J. V. Villumsen, J. W. Wadsley, M. S. Warren, G. Xu and G. Yepes (1999). The Santa Barbara cluster comparison project: a comparison of cosmological hydrodynamics solutions. *astro-ph/9906160*.

Friedmann, W. (1917). *Mon. Not. R. astr. Soc.* **78**, 3+.

Gardner, J. P., R. M. Sharples, C. S. Frenk and B. E. Carrasco (1997). A wide-field k-band survey: The luminosity function of galaxies. *Astrophys. J. Lett.* **480**, L99+.

Gerritsen, J. P. E. (1997). *PhD Thesis*. Kapetyn Astronomical Institute.

Gerritsen, J. P. E. and V. Icke (1997). Star formation in n-body simulations. I - The impact of the stellar ultraviolet radiation on star formation. *Astron. Astrophys.* **325**, 972–986.

Gingold, R. A. and J. J. Monaghan (1977). Smoothed particle hydrodynamics - theory and application to non-spherical stars. *Mon. Not. R. astr. Soc.* **181**, 375–389.

Gingold, R. A. and J. J. Monaghan (1983). On the fragmentation of differentially rotating clouds. *Mon. Not. R. astr. Soc.* **204**, 715–733.

Goldstein, H. (1980). *Classical Mechanics*. Addision Wesley. New York.

Gregg, M. D. and M. J. West (1998). Galaxy disruption as the origin of intracluster light in the coma cluster of galaxies. *Nature* **396**, 549–552.

Groth, E. J., P. J. E. Peebles, M. Seldner and R. M. Soneira (1977). The clustering of galaxies. *Scientific American* **237**, 76–78.

Guth, A. and S.-Y. Pi (1982). *Phys. Rev. Lett.* **49**, 1110+.

Hawking, S. W. (1982). *Phys. Lett. B* **115**, 295+.

Hawley, J. F., J. R. Wilson and L. L. Smarr (1984). A numerical study of nonspherical black hole accretion. I - Equations and test problems. *Astrophys. J.* **277**, 296–311.

Henon, M. (1964). L'evolution initiale d'un amas spherique. *Ann. Astrophys. (Fr.)* **27**, 83–91.

Hernquist, L. (1990). Vectorization of tree traversals. *J. Comp. Phys.* **87**, 137–147.

Hernquist, L. and N. Katz (1989). TREESPH - A unification of SPH with the hierarchical tree method. *Astrophys. J. Suppl. Series* **70**, 419–446.

Heron, A. and J. C. Adam (1989). Particle code optimization on vector computers. *J. Comp. Phys.* **85**, 284+.

Hindmarsh, M. (1998). Cosmic strings - dead again?. In: *COSMO-97, First International Workshop on Particle Physics and the Early Universe : September 15-19, 1997* (L. Roszkowski, Ed.). World Scientific. New Jersey. pp. 420+.

Hockney, R. H. (1966). Computer experiment of anomalous diffusion. *Phys. Fluids* **9**, 1826–1835.

Hockney, R. H. (1967). Gravitational experiments with a cylindrical galaxy. *Astrophys. J.* **150**, 797–806.

Hockney, R. H. and D. R. K. Brownrigg (1974). Effects of population II stars and three-dimensional motion on spiral structure. *Mon. Not. R. astr. Soc.* **167**, 351–357.

Hockney, R. W. and J. W. Eastwood (1988). *Computer simulation using particles*. Adam Hilger. Bristol.

Hoerner, S. von (1960). Die numerische integration des n-korper-problemes fur sternhaufen i. *Z. Astrophys.* **50**, 184–214.

Hohl, F. (1971). Numerical experiments with a disk of stars. *Astrophys. J.* **168**, 343–359.

Hohl, F. and R. H. Hockney (1969). A computer model of a disk of stars. *J. Comp. Phys.* **4**, 306–323.

Horowitz, E. J. (1987). Vectorizing the interpolation routines of particle-in-cell codes. *J. Comp. Phys.* **68**, 56+.

Hubble, E. P. (1926). Extragalactic nebulae.. *Astrophys. J.* **64**, 321–369.

Hubble, E. P. (1929). A relation between distance and radial velocity among extra-galactic nebulae. *Proc. Nat. Acad. Sci.* **15**, 168–173.

Hultman, J. and A. Pharasyn (1999). Hierarchical, dissipative formation of elliptical galaxies: Is thermal instability the key mechanism?. Hydrodynamical simulations including supernova feedback, multi-phase gas and metal enrichment in CDM: structure and dynamics of elliptical galaxies. *Astron. Astrophys.* **347**, 769–798.

Hultman, J. and D. Kaellander (1997). An SPH code for galaxy formation problems. presentation of the code. *Astron. Astrophys.* **324**, 534–548.

Huss, A., B. Jain and M. Steinmetz (1999). How universal are the density profiles of dark halos?. *Astrophys. J.* **517**, 64–69.

Jeans, J. H. (1928). *Astronomy and cosmogony*. Cambridge University Press. Cambridge.

Jenkins, A., C. S. Frenk, F. R. Pearce, P. A. Thomas, J. M. Colberg, S. D. M. White, H. M. P. Couchman, J. A. Peacock, G. Efstathiou and A. H. Nelson (1998). Evolution of structure in cold dark matter universes. *Astrophys. J.* **499**, 20+.

Katz, N. (1992). Dissipational galaxy formation. II - Effects of star formation. *Astrophys. J.* **391**, 502–517.

Katz, N., D. H. Weinberg and L. Hernquist (1996). Cosmological simulations with TREESPH. *Astrophys. J. Suppl. Series* **105**, 19+.

Katz, N., D. H. Weinberg and L. Hernquist (1998). The clustering of high redshift galaxies in the cold dark matter scenario. *astro-ph/9806257*.

Kauffmann, G., S. D. M. White and B. Guiderdoni (1993). The formation and evolution of galaxies within merging dark matter halos. *Mon. Not. R. astr. Soc.* **264**, 201+.

Kay, S. T., F. R. Pearce, A. Jenkins, C. S. Frenk, S. D. M. White, P. A. Thomas and H. M. P. Couchman (1999). Parameter tests within cosmological simulations of galaxy formation. *astro-ph/9908107*.

Kennicutt, R. C., Jr. (1998). The global Schmidt Law in star-forming galaxies. *Astrophys. J.* **498**, 541+.

King, I. R. (1972). Density data and emission measure for a model of the coma cluster. *Astrophys. J. Lett.* **174**, L123+.

Klypin, A., S. Gottloumlber, A. V. Kravtsov and A. M. Khokhlov (1999). Galaxies in N-body simulations: Overcoming the overmerging problem. *Astrophys. J.* **516**, 530–551.

Kolb, E. W. and M. S. Turner (1990). *The Early Universe*. Frontiers in Physics. Addison-Wesley. Reading, MA.

Kravtsov, A. V., A. A. Klypin and A. M. Khokhlov (1997). Adaptive refinement tree: A new high-resolution N-body code for cosmological simulations. *Astrophys. J. Suppl. Series* **111**, 73+.

Lacey, C. and S. Cole (1994). Merger rates in hierarchical models of galaxy formation. II - Comparison with N-body simulations. *Mon. Not. R. astr. Soc.* **271**, 676+.

Leasure, B. (1994). Parallel processing model for high level programming languages. *Draft Proposed American National Standard for Information Processing Systems*.

Lemaitre, G. (1927). *Ann. Soc. Sci. Bruxelles* **47A**, 49+.

Li, X., P. Lu, J. Schaeffer, J. Shillington, P. Wong and H. Shi (1993). On the versatility of parallel sorting by regular sampling. *Parallel Computing* **19**, 1079–1103.

Lifshitz, E. M. (1946). *J. Phys.* **10**, 116+.

Lilly, S. J., O. Le Fevre, F. Hammer and D. Crampton (1996). The Canada-France redshift survey: The luminosity density and star formation history of the universe to z approximately 1. *Astrophys. J. Lett.* **460**, L1+.

Lombardi, J. C., A. Sills, F. A. Rasio and S. L. Shapiro (1998). Tests of spurious transport in smoothed particle hydrodynamics. *astro-ph/9807290*.

Lowenthal, J. D., D. C. Koo, R. Guzman, J. Gallego, A. C. Phillips, S. M. Faber, N. P. Vogt, G. D. Illingworth and C. Gronwall (1997). Keck spectroscopy of redshift z approximately 3 galaxies in the Hubble Deep Field. *Astrophys. J.* **481**, 673+.

Lucy, L. B. (1977). A numerical approach to the testing of the fission hypothesis. *Astron. J* **82**, 1013–1024.

Mac Low, M. M. and A. Ferrara (1999). Starburst-driven mass loss from dwarf galaxies: Efficiency and metal ejection. *Astrophys. J.* **513**, 142–155.

Macfarland, T., H. M. P. Couchman, F. R. Pearce and J. Pichlmeier (1998). A new parallel P3M code for very large-scale cosmological simulations. *New Astronomy* **3**, 687–705.

Madau, P., L. Pozzetti and M. Dickinson (1998). The star formation history of field galaxies. *Astrophys. J.* **498**, 106+.

Malhotra, S. (1995). The vertical distribution and kinematics of H I and mass models of the galactic disk. *Astrophys. J.* **448**, 138+.

Mao, S., H. J. Mo and S. D. M. White (1998). The evolution of galactic discs. *Mon. Not. R. astr. Soc.* **297**, L71–L75.

Martel, H. and P. R. Shapiro (1998). Explosions during galaxy formation: Scale-free simulations. In: *Abstracts of the 19th Texas Symposium on Relativistic Astrophysics and Cosmology, held in Paris, France, Dec. 14-18, 1998* (J. Paul, T. Montmerle and E. Aubourg, Eds.). CEA Saclay. Paris. pp. E524+.

McGaugh, S. (1998). How galaxies don't form. In: *'Galaxy Dynamics' conference held at Rutgers University* (D. R. Merritt, M. Valluri and J. A. Sellwood, Eds.). ASP Conference Series. San Francisco. pp. E30+.

McKee, C. F. and J. P. Ostriker (1977). A theory of the interstellar medium - three components regulated by supernova explosions in an inhomogeneous substrate. *Astrophys. J.* **218**, 148–169.

Mihos, J. C. and L. Hernquist (1994). Star-forming galaxy models: Blending star formation into TREESPH. *Astrophys. J.* **437**, 611–624.

Milgrom, M. (1983*a*). A modification of the Newtonian dynamics - implications for galaxy systems. *Astrophys. J.* **270**, 384+.

Milgrom, M. (1983*b*). A modification of the Newtonian dynamics as a possible alternative to the hidden mass hypothesis. *Astrophys. J.* **270**, 365–370.

Mobarry, C. and J. Crawford (1996). Convex SPP-1000 Exemplar implementation of parallel sorting by regular sampling. *http://outside.gsfc.nasa.gov/ESS/eazydir/inhouse/mobarry/sorting/convex.html*.

Monaghan, J. J. (1988). An introduction to SPH. *Comp. Phys. Comm.* **48**, 89–96.

Monaghan, J. J. (1992). Smoothed particle hydrodynamics. *Ann. Rev. Astron. Astrophys.* **30**, 543–574.

Monaghan, J. J. and J. C. Lattánzio (1985). A refined particle method for astrophysical problems. *Astron. Astrophys.* **149**, 135–143.

Monaghan, J. J. and R. A. Gingold (1983). Shock simulation by the particle method SPH. *J. Comp. Phys.* **52**, 374–389.

Moore, B., F. Governato, T. Quinn, J. Stadel and G. Lake (1998). Resolving the structure of cold dark matter halos. *Astrophys. J. Lett.* **499**, L5+.

Moore, B., N. Katz and G. Lake (1996). On the destruction and overmerging of dark halos in dissipationless N-body simulations. *Astrophys. J.* **457**, 455+.

Moore, B., T. Quinn, F. Governato, J. Stadel and G. Lake (1999). Cold collapse and the core catastrophe. *astro-ph/9903164*.

Navarro, J. F. and M. Steinmetz (1997). The effects of a photoionizing ultraviolet background on the formation of disk galaxies. *Astrophys. J.* **478**, 13+.

Navarro, J. F. and S. D. M. White (1993). Simulations of dissipative galaxy formation in hierarchically clustering universes. I - Tests of the code. *Mon. Not. R. astr. Soc.* **265**, 271+.

Navarro, J. F. and S. D. M. White (1994). Simulations of dissipative galaxy formation in hierarchically clustering universes-2. dynamics of the baryonic component in galactic halos. *Mon. Not. R. astr. Soc.* **267**, 401–412.

Navarro, J. F. and W. Benz (1991). Dynamics of cooling gas in galactic dark halos. *Astrophys. J.* **380**, 320–329.

Navarro, J. F., C. S. Frenk and S. D. M. White (1996*a*). The structure of cold dark matter halos. *Astrophys. J.* **462**, 563+.

Navarro, J. F., C. S. Frenk and S. D. M. White (1997). A universal density profile from hierarchical clustering. *Astrophys. J.* **490**, 493+.

Navarro, J. F., V. R. Eke and C. S. Frenk (1996*b*). The cores of dwarf galaxy halos. *Mon. Not. R. astr. Soc.* **283**, L72–L78.

Nishiguchi, A., S. Orii and T. Yabe (1985). Vector calculation of particle code. *J. Comp. Phys.* **61**, 519+.

Nusser, A. and R. K. Sheth (1999). Mass growth and density profiles of dark matter halos in hierarchical clustering. *Mon. Not. R. astr. Soc.* **303**, 685–695.

OpenMP Consortium (1998). OpenMP application program interface (API). *http://www.openmp.org*.

Owen, J. M. and J. V. Villumsen (1997). Baryons, dark matter, and the Jeans mass in simulations of cosmological structure formation. *Astrophys. J.* **481**, 1+.

Pearce, F. R., A. R. Jenkins, C. S. Frenk, J. M. Colberg, S. D. M. White, P. A. Thomas, H. M. P. Couchman, J. A. Peacock and G. Efstathiou (1999). A simulation of galaxy formation and clustering. *astro-ph/9905160*.

Pearce, F. R. and H. M. P. Couchman (1997). Hydra: a parallel adaptive grid code. *New Astronomy* **2**, 411–427.

Peebles, P. J. E. (1993). *Principles of physical cosmology*. Princeton University Press. Princeton, NJ.

Pettini, M., M. Kellogg, C. C. Steidel, M. Dickinson, K. L. Adelberger and M. Giavalisco (1998). Infrared observations of nebular emission lines from galaxies at z approximately 3. *Astrophys. J.* **508**, 539–550.

Porter, D. (1985). *PhD Thesis*. University of California at Berkeley.

Primack, J. R., J. S. Bullock, A. A. Klypin and A. V. Kravtsov (1998). Formation of dark matter halos. In: *'Galaxy Dynamics' conference held at Rutgers University* (D. R. Merritt, M. Valluri and J. A. Sellwood, Eds.). ASP Conference Series. San Francisco.

Quinn, P. J. (1984). On the formation and dynamics of shells around elliptical galaxies. *Astrophys. J.* **279**, 596–609.

Quinn, T., N. Katz and G. Efstathiou (1996). Photoionization and the formation of dwarf galaxies. *Mon. Not. R. astr. Soc.* **278**, L49–L54.

Rasio, F. A. and S. L. Shapiro (1991). Collisions of giant stars with compact objects - hydrodynamical calculations. *Astrophys. J.* **377**, 559–580.

Rees, M. J. and J. P. Ostriker (1977). Cooling, dynamics and fragmentation of massive gas clouds - clues to the masses and radii of galaxies and clusters. *Mon. Not. R. astr. Soc.* **179**, 541–559.

Salopek, D. S. (1994). *Int. J. Mod. Phys. D* **3**, 257+.

Schuessler, I. and D. Schmitt (1981). Comments on smoothed particle hydrodynamics. *Astron. Astrophys.* **97**, 373–379.

Schwarz, M. P. (1981). The response of gas in a galactic disk to bar forcing. *Astrophys. J.* **247**, 77–88.

Schwarz, M. P. (1984). The intrinsic shape of rings in disk galaxies. *Astron. Astrophys.* **133**, 222–224.

Sellwood, J. A. and E. M. Moore (1999). On the formation of disk galaxies and massive central objects. *Astrophys. J.* **510**, 125–135.

Serna, A., J. M. Alimi and J. P. Chieze (1996). Adaptive smoothed particle hydrodynamics and particle-particle coupled codes: Energy and entropy conservation. *Astrophys. J.* **461**, 884+.

Shandarin, S. F. a[illegible]a. B. Zel'dovich (1989). *Rev. Mod. Phys.* **61**, 185+.

Silicon Graphics Incorporated (1997). *MIPSPro Fortran 77 Programmer's Guide.* SGI document 007-0711-060.

Silk, J. (1997). Feedback, disk self-regulation, and galaxy formation. *Astrophys. J.* **481**, 703+.

Silk, J. (1998). The IMF: Long ago and far away. In: *ASP Conf. Ser. 142: The Stellar Initial Mass Function (38th Herstmonceux Conference).* pp. 177+.

Smoot, G. F., C. L. Bennett, A. Kogut, E. L. Wright, J. Aymon, N. W. Boggess, E. S. Cheng, G. De Amici, S. Gulkis, M. G. Hauser, G. Hinshaw, P. D. Jackson, M. Janssen, E. Kaita, T. Kelsall, P. Keegstra, C. Lineweaver, K. Loewenstein, P. Lubin, J. Mather, S. S. Meyer, S. H. Moseley, T. Murdock, L. Rokke, R. F. Silverberg, L. Tenorio, R. Weiss and D. T. Wilkinson (1992). Structure in the cobe differential microwave radiometer first-year maps. *Astrophys. J. Lett.* **396**, L1–L5.

Sod, G. A. (1978). A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J. Comp. Phys.* **27**, 1–31.

Somerville, R. S. and J. R. Primack (1998). Semi-analytic modelling of galaxy formation: The local universe. *astro-ph/9802268.*

Sommer-Larsen, J., H. Vedel and U. Hellsten (1998*a*). The structure of isothermal, self-gravitating, stationary gas spheres for softened gravity. *Astrophys. J.* **500**, 610+.

Sommer-Larsen, J., S. Gelato and H. Vedel (1998*b*). Formation of disk galaxies: feedback and the angular momentum problem. *astro-ph/9801094.*

Starobinskii, A. A. (1982). *Phys. Lett. B* **117**, 175+.

Steidel, C., K. Adelberger, M. Giavalisco, M. Dickinson, M. Pettin and M. Kellogg (1998). Galaxies and large scale structure at z approximately 3. In: *The Young Universe: Galaxy Formation and Evolution at Intermediate and High Redshift* (S. D'Odorico, A. Fontana and E. Giallongo, Eds.). ASP Conference Series. San Francisco. pp. 428+.

Steinmetz, M. (1996). Grapesph: cosmological smoothed particle hydrodynamics simulations with the special-purpose hardware grape. *Mon. Not. R. astr. Soc.* **278**, 1005–1017.

Steinmetz, M. and E. Mueller (1993). On the capabilities and limits of smoothed particle hydrodynamics. *Astron. Astrophys.* **268**, 391–410.

Steinmetz, M. and E. Mueller (1994). The formation of disk galaxies in a cosmological context: Populations, metallicities and metallicity gradients. *Astron. Astrophys.* **281**, L97–L100.

Steinmetz, M. and E. Muller (1995). The formation of disc galaxies in a cosmological context: Structure and kinematics. *Mon. Not. R. astr. Soc.* **276**, 549–562.

Steinmetz, M. and J. F. Navarro (1999). The cosmological origin of the Tully-Fisher relation. *Astrophys. J.* **513**, 555–560.

Sugimoto, D., Y. Chikada, J. Makino, T. Ito, T. Ebisuzaki and M. Umemura (1990). A special purpose computer for gravitational many-body problems. *Nature* **345**, 33+.

Sutherland, R. S. and M. A. Dopita (1993). Cooling functions for low-density astrophysical plasmas. *Astrophys. J. Suppl. Series* **88**, 253–327.

Thacker, R. J. (1997). Simulations of galaxy formation and clustering. In: *Keynote Seminar, Canadian Astronomical Society meeting, Edmonton, June 1997.*

Thacker, R. J., H. M. P. Couchman and F. R. Pearce (1998). Simulating galaxy formation on smps. In: *High Performance Computing Systems and Applications 1998* (J. Schaeffer and R. Urau, Eds.). Kluwer Academic Publishers. Dordrecht. pp. 213+.

Theuns, T. (1994). Parallel P3M with exact calculation of short range forces. *Comp. Phys. Comm.* **78**, 238–246.

Thomas, P. A. and H. M. P. Couchman (1992). Simulating the formation of a cluster of galaxies. *Mon. Not. R. astr. Soc.* **257**, 11–31.

Thoul, A. A. and D. H. Weinberg (1996). Hydrodynamic simulations of galaxy formation. II - Photoionization and the formation of low-mass galaxies. *Astrophys. J.* **465**, 608+.

Tissera, P. B. and R. Dominguez-Tenreiro (1998). Dark matter halo structure in CDM hydrodynamical simulations. *Mon. Not. R. astr. Soc.* **297**, 177–194.

Toomre, A. and J. Toomre (1972). Galactic bridges and tails. *Astrophys. J.* **178**, 623–666.

Truelove, J. K., R. I. Klein, C. F. MCKee, J. H. Holliman, L. H. Howell II, J. A. Greenough and D. T. Woods (1998). Self-gravitational hydrodynamics with three-dimensional adaptive mesh refinement: Methodology and applications to molecular cloud collapse and fragmentation. *Astrophys. J.* **495**, 821+.

Tully, R. B. and J. R. Fisher (1977). A new method of determining distances to galaxies. *Astron. Astrophys.* **54**, 661–673.

Turok, N. (1996). Sub-degree scale microwave anistropies from cosmic defects. *Astrophys. J. Lett.* **473**, L5+.

U. S. Department of Energy (1996). Accelerated strategic computing initiative (ASCI). *http://www.llnl.gov/asci/.*

Van Den Bosch, F. C. (1998). The formation of disk-bulge-halo systems and the origin of the Hubble sequence. *Astrophys. J.* **507**, 601–614.

Van Kampen, E., R. Jimenez and J. A. Peacock (1998). The Tully-Fischer relation in phenomenological models of galaxy formation. In: *Abstracts of the 19th Texas Symposium on Relativistic Astrophysics and Cosmology, held in Paris, France, Dec. 14-18, 1998* (J. Paul, T. Montmerle and E. Aubourg, Eds.). CEA Saclay. Paris. pp. E584+.

Vedel, H., U. Hellsten and J. Sommer-Larsen (1994). Formation of disc galaxies in the presence of a background UVX radiation field. *Mon. Not. R. astr. Soc.* **271**, 743+.

Viana, P. T. P. and A. R. Liddle (1996). The cluster abundance in flat and open cosmologies. *Mon. Not. R. astr. Soc.* **281**, 323+.

Wadsley, J. (1997). Private communication.

Warren, M. S., J. K. Salmon, D. J. Becker, M. P. Goda and T. Sterling (1997). A treecode at 430 gigaflops on ASCI Red, Part I of 1997 Gordon Bell performance prize. *Gordon Bell Performance Prize Winner.*

Weil, M. L., V. R. Eke and G. Efstathiou (1998). The formation of disc galaxies. *Mon. Not. R. astr. Soc.* **300**, 773–789.

Weinberg, D. H., L. Hernquist and N. Katz (1997). Photoionization, numerical resolution, and galaxy formation. *Astrophys. J.* **477**, 8+.

Weinberg, S. (1972). *Gravitation and Cosmology: Principles and Applications of the General Theory of Relativity.* Wiley. New York.

White, S. D. M. (1984). Angular momentum growth in protogalaxies. *Astrophys. J.* **286**, 38–41.

White, S. D. M. (1994). Formation and evolution of galaxies: Les Houches lectures. *astro-ph/9410043.*

White, S. D. M. and C. S. Frenk (1991). Galaxy formation through hierarchical clustering. *Astrophys. J.* **379**, 52–79.

White, S. D. M. and M. J. Rees (1978). Core condensation in heavy halos - a two-stage theory for galaxy formation and clustering. *Mon. Not. R. astr. Soc.* **183**, 341–358.

Wood, D. (1981). Collapse and fragmentation of isothermal gas clouds. *Mon. Not. R. astr. Soc.* **194**, 201–218.

Yepes, G., R. Kates, A. Khokhlov and A. Klypin (1997). Hydrodynamical simulations of galaxy formation: effects of supernova feedback. *Mon. Not. R. astr. Soc.* **284**, 235–256.